POLICY APPLICATION TO RTI POOLER NETWORKS FOR OPTIMIZED RELOCATION PLANNING

Optimizing RTI pooling management for reducing operational costs and increasing sustainability in supply chains

MSc thesis Engineering and Policy Analysis

Keywords: Sustainability, network flow problem, sensitivity analysis, policy

TU Delft Delft University of Technology

Jasper Endlich
4286995

Delft University of Technology
Master Thesis

Policy application to RTI pooler networks for optimized relocation planning

---

Master thesis submitted to
Delft University of Technology

in partial fulfilment of the requirements for the degree of
Master of Science in Engineering and Policy Analysis

Faculty of Technology, Policy and Management

by

Jasper Endlich
4286995

Graduation committee

Chairperson	Prof. Dr. Rolf Kunneke, Faculty of Technology, Policy and Management, Delft University of Technology
First Supervisor	Dr. Stefano Fazi, Faculty of Technology, Policy and Management, Delft University of Technology
Second Supervisor	Prof. Dr. Rolf Kunneke, Faculty of Technology, Policy and Management, Delft University of Technology
External Supervisor	Bec. Erik Burgers, Container Centralen AS

Summary

In this thesis, the optimization of inventory management within RTI pooler networks is considered. The inventory management by RTI poolers consists of the network configurations on which the network relies, as well as the relocation schedule of RTI poolers that determines the functionality of the system.

To optimize both the RTI pooler network and its relocation schedule, the algorithmic framework approach of Lau et al. (2000) is applied. Their research considers the use of a master-slave fashioned framework to solve an inventory routing problem using a network flow problem and a vehicle routing problem. In this thesis, the multi-depot inventory routing problem is solved through the use of a network flow problem as the master and a multi-knapsack problem as the slave. For the network flow problem, the aim is to develop a minimum cost and minimum distance flow schedule that satisfies all the deterministic demand, based on the model developed by Karsten et al (2015). For the second part of the algorithmic framework, a multi-knapsack problem is solved in order to determine the minimum number of vehicles required for the relocation schedule. The RTI pooler's network flow problem is thus expanded with a multi-knapsack problem in a master-slave algorithmic framework.

In order to find the optimal schedule, several mixed integer linear programming formulations have been formulated for the algorithms used in this thesis. These mathematical formulations are aiming to optimize the schedule based on the objectives of RTI poolers; maximizing sustainability, cost-efficiency and functionality of their RTI pooler management. Based on the preferences of the RTI poolers, the trade-off between these three objectives can be formalized for which the optimal solution is determined. As the solution is heavily dependent on the given weights to each of the objectives, it is crucial to determine the importance of each of the objectives with respect to the trade-off for RTI poolers.

Next to finding the optimal relocation schedule, the effects of changes to the RTI pooler network in the form of applying policies can be measured through the use of these models. Policies such as alternate depot capacities, additional repair locations and acquiring additional inventory can be applied to RTI pooler networks. The models have been applied to a case study at the large RTI pooler in the horticultural sector; Container Centralen AS, where the models have shown the effectivity of the optimized scheduler as well as the effects of applying policies to alter RTI pooler networks. With the use of the mentioned case study, the positive effects of applying additional policies to the network flow problem are identified. Giving the RTI pooler the opportunity to implement one or multiple policies in order to achieve their objectives set.

Further research could involve improving the quality of the mathematical formulations by merging the two models from the master-slave algorithmic framework into one feasible mathematical model. The scope of the model can be expanded by the introduction of additional policies next to the policies mentioned in this thesis, such as price incentives or switching to a hub-spoke type network. Finally, the effects of simultaneously applying multiple policies to RTI pooler network can be studied further in order to achieve the most optimal solution possible for any RTI pooler network.

Preface

After many iterations and starting over multiple times during long days and late hours, all the sections finally fall into place. Overall, I am proud of what I have achieved during the course of this thesis.

First of all, I would like to thank my graduation committee from TU Delft and Container Centralen AS. Thank you Stefano for assisting me from the start with building up my thesis, your guidance has brought me and this thesis to where it is now. Especially the mathematical elements of this thesis have only been possible because of you. Thank you Rolf for your sharp eye and keeping me focused on the policy aspect of my thesis. It has helped me to be more specific and sharper in my writing. Finally, I would like to thank Erik for your continuous support and enthusiasm, as well as the trust in me throughout the process.

I would like to thank everyone I worked with at Container Centralen for their assistance and help with mapping the RTI pooler network of their company. Thank you Eldin, Jens, Jordi, Andrea and many others for always being open to assist me and clarify any of my questions.

Finally, a big thank you to my family, girlfriend and friends for keeping me sane during these last few months when I needed them.

Yours sincerely,

Jasper Endlich

September 16, 2021

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CLSC	Closed-loop supply chain
IRP	Inventory routing problem
MCNFP	Minimum cost network flow problem
MDIRP	Multi-depot inventory routing problem
MILP	Mixed integer linear programming
ML	Maximum level
NFP	Network flow problem
OUL	Order-up-to level
RTI	Reusable transport item
SCM	Supply chain management

1. Introduction

Driven by both financial as well as sustainability motives, innovation and optimization within the logistics sector has been taking place in the last decades. Using Reusable Transport Items (RTIs) for transporting goods has been one of these innovations and has been adopted in numerous supply chains. The use of RTIs within transport drastically reduces emissions, as they can be used multiple times by different actors (Elia et al., 2015). Many forms of RTIs are known such as pallets, barrels, trolleys, kegs and refillable gas containers (ISO, 2007), but independent of the form of the RTIs themselves, their sustainable characteristics are shared among all of them. These environmental benefits make the use of RTIs within supply chains extremely interesting but does bring extra challenges in the form of reverse logistics and policies for RTI management. Because of the improved availability and communication of data, the RTI pooler has become an important player within supply chains. RTI poolers give other companies the opportunity to freely use RTIs for transport, by taking both control and responsibility of the forward and reverse logistics that are associated with the use of RTIs.

Multiple depots, managed by the RTI pooler, are scattered over a large area and form the pooler's network. At these depots, an in- and outflow of RTIs is causing the inventory level to fluctuate over time, as it can be seen in the figure below. Combining the in- and output flows of all of the depots within the network, the result is a continuously active circular network where RTIs flow freely. To maintain balance within the system, homogeneous capacitated vehicles are deployed for relocating stock. These vehicles, responsible for both pick-up and delivery of either working or broken stock, must depart from their origin depot and may end their route at any other depot. Relocation of working stock can be initiated to any other depot to meet demand, while broken stock needs to be transported to the repair location before re-entering the system. An overview of what the input and output of depots looks like is given in Figure 1.

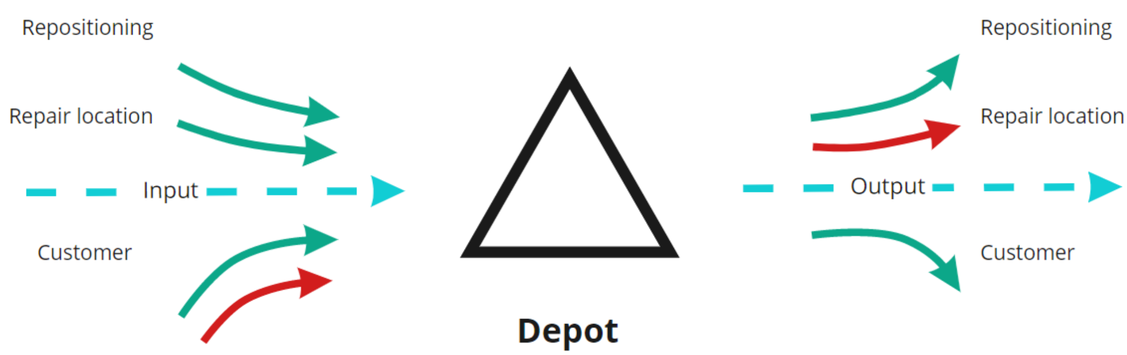

Figure 1. Input and output flows of depots

Since the network consists of both forward and reverse supply chains, it can be labeled as a closed-loop supply chain network. The main operation for the RTI pooler within this network concerns the inventory management at the depots, based on demand and returns rates which are assumed to be deterministic. The goal in this regard is to create a schedule to efficiently relocate both working and broken stock between depots in order to prevent depots experiencing stock-outs while minimizing transportation and handling costs. The overall goal of RTI poolers is to optimize the use of RTIs within logistics, increasing the sustainability of the sector and reducing costs and complexity for other actors involved. Due to the increase in demand for sustainable initiatives across all sectors, RTI poolers are able to fulfill this need and increase the quality of logistics in general to make sure it is future-proof.

The RTI pooler is solely responsible to maintain the system balance and the RTIs within, including the management of the systems network. And ideally, as little operations as possible are required for the system to function efficiently (Xiao et al., 2013). For example, in case of an expected stock-out at a certain depot in the near future, working stock is relocated to that specific depot from a depot with excessive stock.

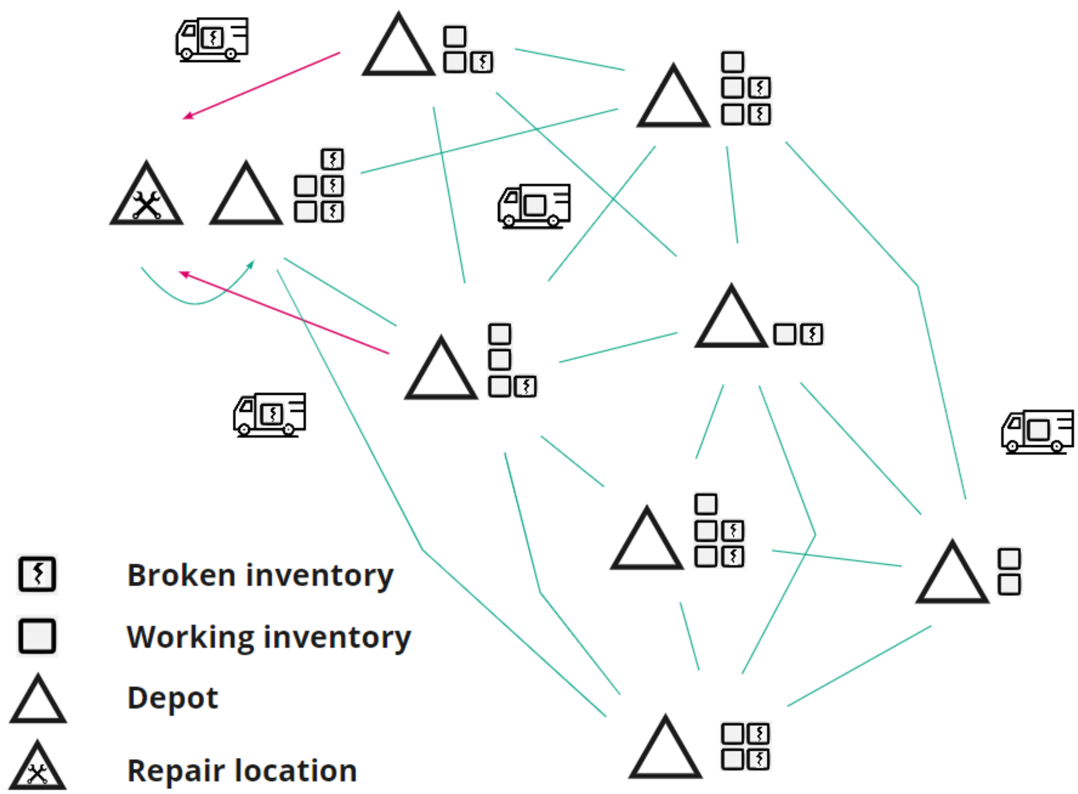

Figure 2. Network design of a pooler system

The proposed problem can be classified as a network flow problem over time, as defined by Fleischer and Skutella (2002). Network flow problems (NFPs) have been studied extensively in scientific literature, starting with the first network flow problems defined by Ford and Fulkerson (1966). Both static and dynamic network flow problems, with ranging objectives from maximum flow to minimum costs, have been introduced in their work. Based on the network flow problems described by Ford and Fulkerson (1966), Karsten (2015) presents and solves a minimum cost dynamic network flow problem which can be applied to other research that involves deterministic demand over time. Their objective is to determine which of the available items to ship on which routes, given a network regarding multiple commodities. Due to the deterministic nature of the RTI pooler network, the work of Karsten can be applied to optimize the inventory management of RTI pooler networks. However, in this thesis we address a network flow problem used for inventory management where several policy applications are quantified and optimized. The result of the network flow problem is dependent on the policies applied to the RTI network, which is to be optimized as a whole. This combination, to our knowledge, has never been tackled in literature before. This thesis therefore aims to fill this gap.

The aim of this thesis is to model and solve the above described problem and the application of several policies for RTI pooler networks by means of a mathematical linear programming model and a commercial solver. Numerical experiments seek to investigate the complexity of the trade-off between objectives as well as the impact of the different parameters through a sensitivity analysis.

Container Centralen (CC), a large RTI pooling company within the floricultural sector, will provide us with the required data on demand and inventory levels at depots, as well as in-depth knowledge of the system and its characteristics. As CC is seen as the industry standard and is responsible for all elements possibly available in RTI pooler system, their network is especially interesting for numerical analysis. With the experiments we aim to generate managerial insights to achieve RTI pooler goals of maximized efficiency and functionality of the inventory management for increased sustainability with minimized operational costs. Next to managerial insights for the RTI pooler, the goal is to future proof RTI pooler networks to be able to cope with the changing demand towards sustainable initiatives and adhere to the changes in the market within the logistics sector.

1.1 Research question

To address the knowledge gap identified in the literature review, modeling and sensitivity analysis will be performed. This will be done by answering the following research question.

"How to optimize the planning of RTI pooler networks, that seek to maximize sustainability, cost-efficiency and functionality, with respect to different policy applications to the network?"

1.2 Sub questions

In order to answer the main research question, the problem will be researched in a stepwise manner. Answering each of the following sub questions will guide me towards answering the main research question.

1. *“What are the features of inventory management and RTI pooler networks?”*
2. *“How can the efficiency of inventory management by RTI poolers be defined and optimized?”*
3. *"How can the problems and additional policies be formulated in formal mathematical models?"*
4. *“How can these mathematical models be solved, with regards to the goals of RTI poolers?”*
5. *“In the case of Container Centralen, can the objectives of their RTI pooling network be achieved through applying the model and additional policies?”*

1.3 Scope

The main focus of the research will be on the development of a mathematical optimization model and the application of policies on this optimization model. Based on the optimization model, the result will be an optimized relocation process where costs, sustainability and functionality are taken into account. With the addition of policies to this model, RTI poolers are able to address the influence of these policies on their network. The background of inventory management, RTI pooling management and the implementation of policies is presented in the literature review.

The following aspects are within the scope of this research:

- **Reverse logistics of RTI pooler networks**
  RTI pooler networks consist of both forward- and reverse supply chain logistics, but only reverse supply chain logistics are considered in this research. The network layout as well as the RTI poolers activities are therefore optimized, but the customers activities are given as model input.

- **Optimization of relocation process**
  The RTI pooler relocation activities on its network consists of initiating flows between depots, transported by capacitated vehicles. Based on the demand and returns within the network, these flows are initiated to upkeep the inventory levels at all times.

- **Policies regarding the network configuration**
  On top of the optimization of the relocation process within RTI pooler networks, other network elements such as the depots, purchasing decisions and repair locations are regarded and altered in order to improve the network. These policies are added to the main mathematical formulation in order to address their influence on the RTI pooler objectives.

Outside the scope of this research are the following aspects:

- **Forward logistics within RTI pooler networks**
  As mentioned, RTI pooler networks consist of both forward- and reverse supply chain logistics. Since RTI poolers are mostly responsible for the reverse logistics, the forward supply chain logistics are not taken into account in this thesis. For this reason, policies regarding the forward supply chain logistics such as price incentives for returning inventory at a different location are not taken into consideration.

- **Vehicle elements**
  The outcomes of optimizing the relocation process are, among other factors, dependent on the vehicles used for relocating inventory. In order to best interpret the results with and without optimized relocation, the vehicles are outsourced and kept constant. Therefore, vehicle capacities are fixed and there is an infinite number of vehicles available. This causes policies such as the addition of alternative vehicles to be excluded from this research.

1.4 Relevance

Both the relevance for scientific literature as well as the relevance for the business is elaborated below.

Relevance to literature

Using a dynamic network flow problem, the implementation of policy choices within the model has not been researched extensively before, to the best of my knowledge. The combination of modeling with the implementation of policy aspects will increase the complexity of the model but will increase the applicability to several problems in both literature as well as physical problems. Through the use of the master-slave algorithm framework, the model is more applicable to other RTI pooling networks and the effects of additional policies can be quantified. The combination of mathematical modeling with policy implementations has not been researched extensively, especially not within the logistic sector where RTI poolers belong to.

Relevance to business

The use of an optimization model for RTI poolers will help to increase the sustainability, cost-efficiency and functionality of RTI pooling network. The trade-off between sustainability and financial benefits will be addressed using the outcome of the model, looking at the policies that can be applied within RTI pooling networks. The model will address the problems in the way that suits the goals of any RTI pooler, as it makes use of the weighted sum objective function. The effects of additional policies to be implemented by companies can be used to achieve the companies' objectives. The objectives set in Section 4.2 are to be achieved.

Relevance to society

Addressing the societal problem of global warming is essential for the future of life on earth. Since the need for sustainable initiatives is growing, RTI pooler networks are aiming to be increasingly sustainable. This thesis aims to assist the management of RTI pooler networks by quantifying the effects of optimizing both the scheduling as well as implementing policies on the RTI pooler network, in order to increase the sustainability of their company. Through the increased understanding of improving sustainability within RTI pooler networks, society in general reaps the benefits.

1.5 Structure of this thesis

This thesis is structured as follows. In Section 2, relevant literature is reviewed on RTI pooler networks, inventory management and network flow problems. Section 3 presents how the problem at hand is formulated and a mathematical model is developed for it. The additional policies to be implemented by RTI poolers are also given in this section. In Section 4, how the problem is solved including the methods used in this thesis are elaborated. Applying the mathematical model and its policies to a case study, Section 5 shows the results and implementation. Finally, Section 6 and 7 contain the discussion and conclusion based on the results of this thesis.

2. Literature study

Inventory management in network flow problems in RTI pooler networks

In this Section, the relevant scientific literature on RTI pooler management, inventory management and network flow problems is provided. In the first Subsection, the concept of RTI pooler management is introduced. Following the first Section, secondly the background of RTI management is elaborated in Subsection 2.2. Following with the literature on network flow problems in the final Subsection.

2.1 The concept of RTI pooler management

The concept of RTI pooler management consists of inventory management within a pooler network of reusable transport items. Effective RTI management can present environmental, economic and operational benefits to organizations (Liu et al, 2020), compared to one-way packaging. But because of the complexity of managing an RTI system, creating effective RTI management takes a lot of effort and time. Outsourcing RTI management is therefore prioritized by most, creating the opportunity for specified RTI management parties. RTI poolers have been filling this gap, by taking over the RTI management of most companies involved in the supply chain. RTI poolers are thus responsible for parts of the forward supply chain logistics as well as the entire reverse supply chain logistics that are accompanied by the use of RTI's (Accorsi, 2019).

2.2 Background on RTI management

Scientific literature focused on the strategies of RTI management is not vast, although interest has been growing over the last decade. As can be seen in the RTI management literature review by Glock (2017), research has focused on these four areas: comparative studies of RTI systems and strategies, forecasting RTI returns, purchasing decisions of RTIs, and managing RTI systems.

Only recently, comparative studies of RTI systems and strategies have been increasingly popular. Research by Carrano et al. (2015) has compared three types of RTI management, expendable, buy/sell program and leased. Their findings concluded that for all lifecycles of RTIs, the RTI leasing/pooling program is the most environmentally friendly. The research of Accorsi et al. (2019) extends this research, as it compares four different forms of RTI pooling to each other based on their environmental impact, volume and distance. Out of the forms considered in the research, central hub RTI pooling is shown to be promising. But since the research is based on a case study, the specific characteristics of the network do not resemble characteristics of other networks.

From an RTI pooler network point of view, the characteristics of RTI networks have been researched in several instances. Tornese et al. (2018) have researched the impact of loading conditions and repositioning distances on both an economic as an environmental level, where the network design has been proven to influence both of these aspects. The research of Elia and Gnoni (2015) assesses the effect of bundling and postponing the reverse logistics, including the performance of the reverse supply chain. Different network configurations of the RTI poolers network have been compared in their research, just as in research by Amin et al. (2018), where the RTI pooler network with its suppliers, customers, depots and distributors including their characteristics are examined. Where Elia and Gnoni (2015) have made a framework for better understanding of Closed-loop supply chain (CLSC) system, Amin et al (2018) have formulated a generic mathematical optimization model to solve possible network problems in these CLSC systems such as the location of depots. The optimal repair process for RTIs has been researched by Cobb (2016), by looking at deterministic demand and repair rates. Their work presents an inventory control model for RTI management in a CLSC, including both inspection/repair functions and purchase functions. All of which have been shown to affect the sustainability of the network as a whole, which has been addressed in research by Accorsi et al. (2019).

Additional to the research of RTI management above, research by Bowman et al. (2019), Hellstrom et al. (2009) has been looking into providing insights in the rotation and availability of RTIs within the network. Individual container tracking with RFID is promising, but not will be considered in this research due to the complexity of the technique. Finally, Na et al. (2019) calculated the optimal initial amount of RTIs to be purchased for an efficient supply chain. But as the system in this thesis has already been established, looking at purchasing decisions for RTIs is not applicable to this research.

Both Tornese et al. (2018) and Accorsi et al. (2019) propose future research into dynamic repositioning strategies for RTIs, as well as Elia and Gnoni (2015) and Carrano et al. (2015) who propose to test inventory management models on complex situations. This proposed research is the gap that this thesis is aiming to fill.

2.3 Literature on network flow problems

In order to model complex situations where dynamic repositioning is used for inventory management, network flow problems are researched. Kotnyek (2003) considers an overview of dynamic network flows, with regards to solving network flow problems with different objectives. The network flow problem differs based on the objective of the network, such as maximum flow, quickest flow or minimum cost flow. Within these network flow problems, subproblems such as the minimum cost quickest flow problem are more specific and more applicable to the inventory management of RTI pooler networks.

Sifaleras (2013) has researched and presented a variety of the more specific Minimum Cost Network Flow Problems (MCNFP). One of the examples is the minimum cost quickest flow problem, considering lower and upper bounds for all arcs within the network regarding multiple commodities. Just as in this thesis, once the bound of arcs is unlimited, the network flow problem is labeled as an uncapacitated network flow problem. In the research by Sifaleras each node has a certain demand and supply, similar to nodes within RTI pooler networks. The mathematical formulation of Sifaleras will therefore be used as an example because of the use of demand and supply within a network, especially due to the involvement of time in its research.

Network flow problems that involve time are referred to as dynamic network flow problems. Some examples dynamic network flow problems are the maximum flow, quickest flow or, as tackled in this thesis, the minimum cost network flow problem. The difference between static and dynamic networks and how to address and solve them has been explained in research from Kotnyek (2003) and Fleischer and Skutella (2003). Static network flow problems do not include the aspect of time into their models, thus aiming on pathfinding within networks. Including time within static network flow problems creates dynamic network flow problems, which drastically increasing the calculation time. Figure 3 below shows an indication of the static network flow problems within the dynamic network flow problems.

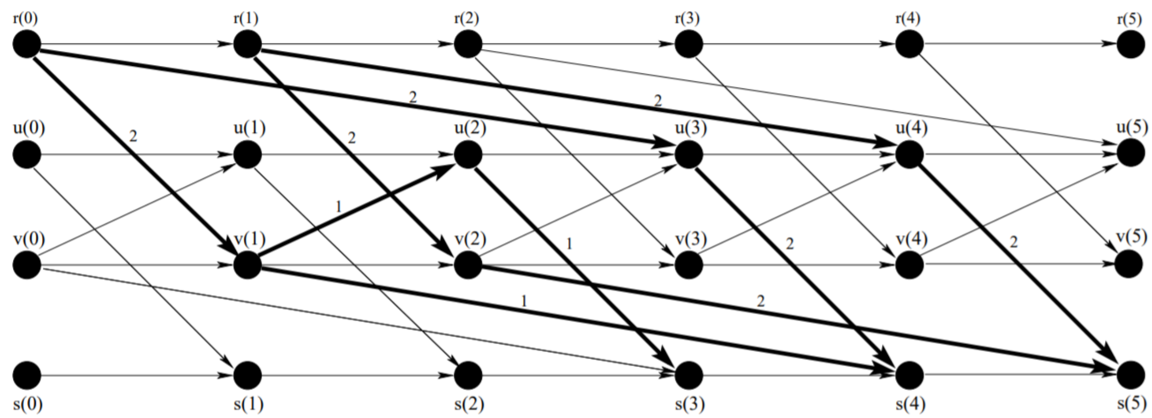

Figure 3. Static vs dynamic network flow

Karsten et al (2015) have formulated a multi-commodity network flow problem to improve the Liner Shipping Network Design. Similar to the network flow problem in this thesis, their aim is to make improvements in the underlying network of the liner shipping company to achieve both environmental and economic benefits. Their research also includes the consequence of neglecting time constraints in existing networks, showing the difference in outcomes between neglecting and including transit time. Where the inclusion of transit time is shown to increase the effectivity of the model.

Building on the inclusion of transit times, research has increasingly included travel time in models. Where Agarwal and Ergun (2008) do include periods and scheduling in their research, they have yet to include travel time restrictions. Which is exactly what Wang and Meng (2011, 2012) and Wang et al. (2013) have done. Their research has initiated modeling considering travel time for the routing within given networks. Their research combined with the mathematical formulation of Alvarez (2011) will be used as a guideline for the implementation of travel time into the model within this thesis.

Concluding, effective RTI management creates the opportunity for the logistics sector to improve its sustainability and cost-efficiency. Due to the introduction of dedicated RTI poolers, the effectivity of RTI management has increased. However, few studies have researched the optimal configuration of RTI pooler networks, including both the network design as well as its relocation schedule. Through the formulation of a minimum cost network flow problem and the addition of policies to the underlying network, this thesis aims to fill this gap in literature.

3. Problem formulation

Inventory management with inventory relocation scheduling, a network flow problem

In the problem formulation below, the main model including its characteristics and details of the problem at hand are elaborated. The problem setting is partially based on the case study, including the characteristics of generic RTI pooling management. After the detailed outline of the problem setting where the assumptions are mentioned and explained, the mathematical models are introduced in the second subsection including the linear programming formulations. Finally, the policies to be implemented in the network are introduced, including its mathematical formulation and policies.

3.1 Formal description of the problem setting

RTI pooler systems are made up out of a set of depots N divided over a large geographical area. Each of these depots $i \in N$ serves RTIs to numerous customers, which can pick up the RTIs at the pooler's depot. Next to the customers collecting RTIs, customers also return RTIs to depots, which are then available for collection again. On top of the collection and returns, these depots are also responsible for exchanging broken RTIs for functional RTIs by customers. These broken RTIs can be repaired at specific repair depots, causing the pooler to initiate transports to bring these broken depots to the repair depot. All this is initiated within the time horizon T. The demand at depots therefore consists of collections and outgoing exchanges, while the returns consists of returns and exchanges.

Because of the demand and returns of RTIs mentioned before, imbalances within the system might occur. These imbalances are in the form of shortages or excesses in inventory at depots within the network, because of the inflow and outflow of items. Based on the availability of information, the opportunity is created to calculate an optimal relocation schedule for optimizing the inventory management, which consists initiating relocations of items between depots. These initiated relocations between depots resemble flows between nodes, corresponding to a network flow problem.

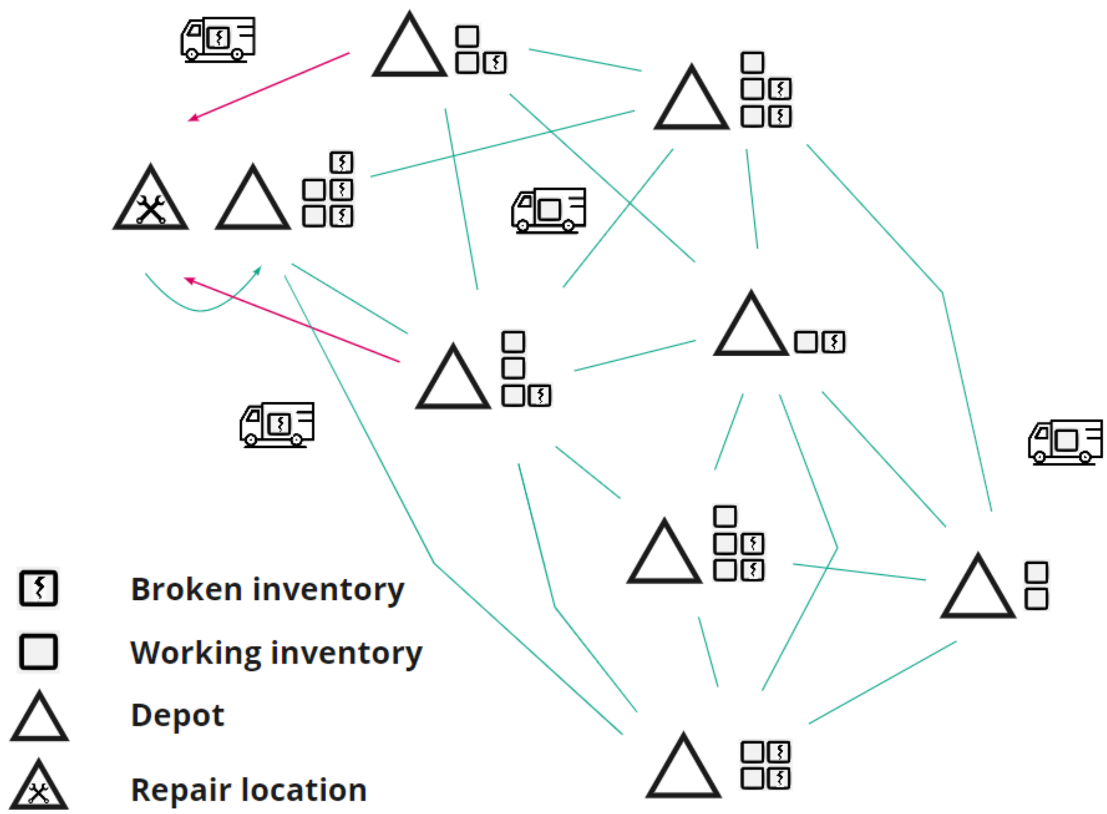

Figure 4. Example of depot network configuration

The RTI pooler's system is made up out of a customer and a pooler side, which can be seen in figure 5. The customer side of the system contains the demand, return and exchange flows and is deterministic, while the pooler side of the system only consists of action of which the pooler has full control. These actions involve (among others) the repositioning of functional RTIs to another depot or the repositioning of broken stock the repair location. The RTIs within the network are made up out of three separate parts which are homogeneous in nature. The vehicles used for repositioning within the system are outsourced to external trucking companies, leaving solely the transportation and handling costs to the RTI pooler.

The goal of the RTI pooler is to manage the inventory levels at depots, by initiating flows of RTIs between depots within the network. The aim is to make sure that within the time horizon, depots do not experience stock-outs while improving the sustainability of the inventory management. As a side objective, the operational costs should be minimized, which consists of transportation costs and handling costs at depots.

Due to the vehicles being outsourced to transportation companies, there is an infinite number of vehicles available for relocation by the RTI pooler. The availability of vehicles also implies is there are no transportation limits to the number of items to be relocated, other than the amount of inventory available at the starting depot. Since the vehicles are outsourced as well, vehicle prices are fixed as can be seen in the second assumption. Therefore, within the model, only the vehicle capacities as well as the pricing per item are included.

Finally, the third and last assumption contains the preparation period as well as the transport time for relocation. Relocations are initiated exactly one period early to account for the loading activities at the starting depot, transport time between depots and the unloading of items at the delivery depot. As relocation is initiated because of forecasted demand at a certain location, the choice for one period of anticipation is verified.

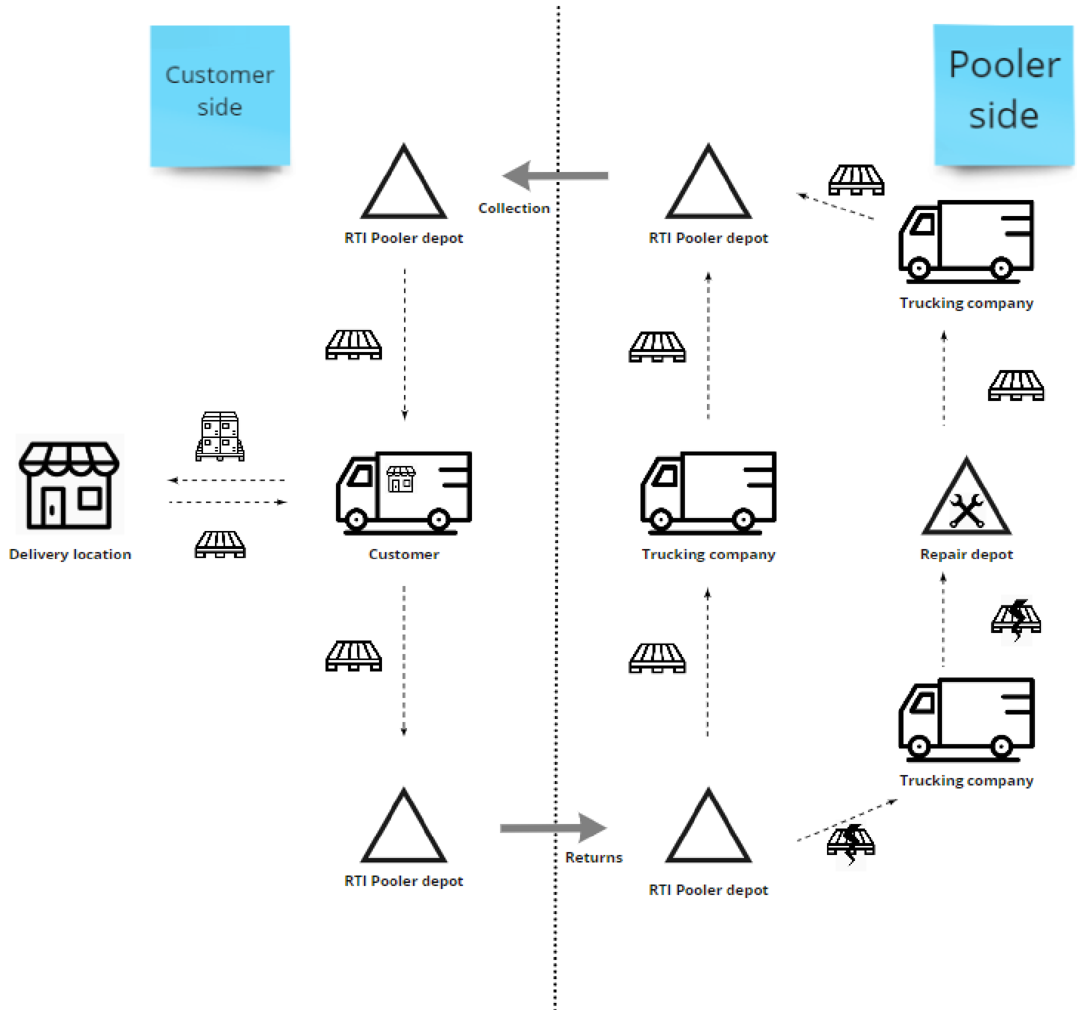

Figure 5. Technical map of actions within the system

3.1.2 Different types of commodities

Within the system, inventory can be divided over multiple commodities. These commodities can be separated such as different types of RTI, or several separate elements that form RTIs once combined. Demand and returns are therefore also specified as demand and returns of one or more commodities from the set of commodities K. An example of a set of commodities K can be seen in Section 5, where the set consists of three items that combinedly form a complete RTI that can be used by customers. In the case study within this thesis, every RTI consists of a single base, four posts and several shelves depending on the size of the items to be transported. Because of the variety of commodities, the inventory definition at depots is determined per commodity within the system.

Next to the different types of inventory, the items can also be divided based on their current state: functional or broken. Functional inventory is relocated to normal depots and available for satisfying the demand of customers, while broken inventory is to be relocated to repair depots in order to be repaired and create more functional stock. Because of the different states of inventory, different inventory formulations are defined for the different states of inventory. The flows of functional and broken items are separated, meaning that they cannot be transported within the same vehicle. At depots the items are stacked separately as well, in order to create a clear division between both sets of inventories.

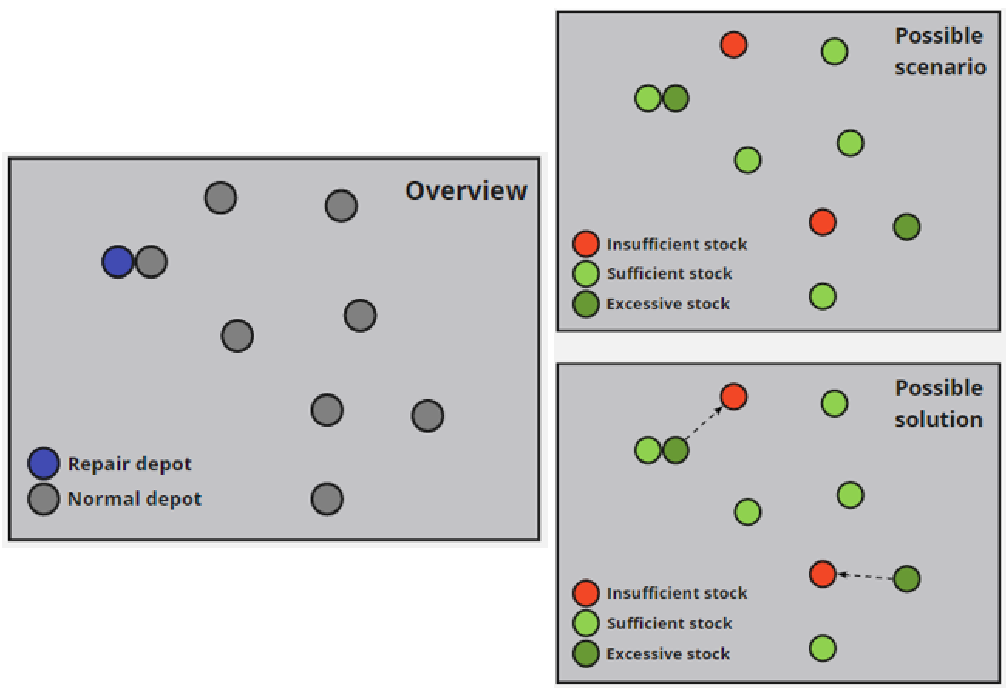

Figure 6. Overview of the system including possibilities

3.2 Mathematical formulation

In the following sections, the mathematical formulations of the situation and problems described in Section 3.1 are given.

3.2.1 The network flow problem

The mathematical model can be found below. It aims to minimize the total relocation distance required for the system to operate, while maintaining or possibly increasing the functionality of the RTI pooler network.

The complete directed graph $G = (N, A)$ is considered, where N is the set of depots and A the set of arcs. The set N consists of all types of depots within the system $N = (N_{depot}, N_{repair})$. This consists of $N_{depot} = \{1, 2, \dots, z\}$ and $N_{repair} = \{z + 1, z + 2, \dots, m\}$. Over a finite and discrete time horizon H, where $T = \{1, 2, \dots, H\}$. Due to the inclusion of time in the model, it is classified as a dynamic network flow model. This causes the network to evolve over time, where the system is different for every time step within the time horizon. Inventory values are likely to be different over time due to the demand and return values that differs over time.

The arc set A is defined as $\{(i, j): i \in N, j \in N, i < j\}$. A non-negative cost C_{ij} is associated with each edge $(i, j) \in A$. As graph G is a Euclidian graph, the triangular inequality holds which means that traveling from i to j directly is always the shortest route. The set of commodities K is used within the system, where all model values are defined per commodity. Each depot $i \in N$ has a demand D_i^t, an inventory I_i^t and a maximum inventory level C_i, where the demand is met through the supply of each depot. If the inventory level at a depot is insufficient and is unable to fulfill the demand, an order y_{ij}^t is created at that depot. These orders are filled by other depots that are able to supply RTIs to other depots. The inventory level at any depot cannot be negative, while stock-outs are not allowed.

The mathematical formulation is based on the following variables:

- I_i^t: the inventory level at depot i at the end of period t
- y_{ij}^t: quantity to deliver to depot i in period t by from depot j

Where the goal is to optimize the function in order to achieve the goal of the RTI pooler.

The Mixed-Integer Linear Programming formulation of the network flow problem is made up using the following sets, parameters and variables.

Table 1. Notations used within the extended network flow model

Sets	Definition
G	The complete directed graph representing the entire system $G = (N, A)$
A	All the arcs within the system, equal to all the different routes between depots
N	The set of all nodes within the system, $N = (N_{depot}, N_{repair})$
N_{depot}	The set of all depots within the system, $N_{depot} = \{1, 2, \dots, z\}$
N_{repair}	The set of all repair locations within the system, $N_{repair} = \{z+1, z+2, \dots, m\}$
T	Set of time including the time horizon H, $T = \{1, 2, \dots, H\}$
K	The set of all commodities within the system, $K = \{K_{working}, K_{broken}\}$
$K_{working}$	The set of all working commodities within the system, $K = \{1, 2, \dots, l\}$
K_{broken}	The set of all broken commodities within the system, $K = \{l+1, l+2, \dots, k\}$
Parameters	
D_{ik}^{t}	Demand at depot i of commodity k at time t, which is deterministic
R_{ik}^{t}	Returns at depot i of commodity k at time t, which is deterministic
Q_{ik}^{t}	Repairs at depot i of commodity k at time t, which is deterministic
D_{ij}	Distance between pick-up depot j and drop-off depot i
C_{ij}	The costs of pick-up at depot j and drop-off at depot i. This is a combined cost that consists of transport and handling cost: $C_{total} = C_{transport} + C_{handling}$
C_{i}^{s}	The cost of going below safety stock at depot i
S_{ik}	Safety stock of commodity k at depot i
U_{ik}	Maximum capacity of stock at depot i per commodity k
P_{k}	The maximum amount of inventory to be introduced of commodity k
Variables	
I_{ik}^{t}	Inventory of commodity k at depot i at time t
y_{ijk}^{t}	Quantity of commodity k to deliver from depot i to depot j in period t
n_{ij}^{t}	Number of vehicles to flow from depot i to depot j in period t
z_{i}^{t}	Binary value that indicates a low stock at depot i in period t

Which are used in the following mathematical formulation of the network flow problem:

$$min \sum_{t \in T} \sum_{(i,j) \in A} \left(\alpha * D_{ij} + \beta * C_{ij}\right) * y_{ijk}^t + \sum_{t \in T} \sum_{i \in N} \gamma * C_i^s * z_i^t \tag{3.1}$$

Minimize

Subject to

$$I_{ik}^t = I_{ik}^{t-1} + R_{ik}^t - D_{ik}^t + Q_{ik}^t + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^t + A_{ik}^t \quad \forall i \in N_{depot}, \forall k \in K_{working}, \forall t \in T \tag{3.2}$$

$$I_{ik}^t = I_{ik}^{t-1} + R_{ik}^t - Q_{ik}^t + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^t \quad \forall i \in N_{depot}, \forall k \in K_{broken}, \forall t \in T \tag{3.3}$$

$$I_{ik}^t = I_{ik}^{t-1} + Q_{ik}^t + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^t \quad \forall i \in N_{repair}, \forall k \in K_{working}, \forall t \in T \tag{3.4}$$

$$I_{ik}^t = I_{ik}^{t-1} - Q_{ik}^t + \sum_{j \in N} y_{jik}^{t-1} \quad \forall i \in N_{repair}, \forall k \in K_{broken}, \forall t \in T \tag{3.5}$$

$$I_{ik}^t \geq S_{ik} - M * z_i^t \quad \forall i \in N_{depot}, \forall k \in K_{working}, \forall t \in T \tag{3.6}$$

$$I_{ik}^t \geq 0 \quad \forall i \in N, \forall k \in K, \forall t \in T \tag{3.7}$$

$$I_{ik}^t \leq U_{ik} \quad \forall i \in N, \forall k \in K, \forall t \in T \tag{3.8}$$

$$\sum_{i \in N} \sum_{k \in K} \sum_{t \in T} A_{ik}^t \leq P_k \tag{3.9}$$

$$z_i^t \in \{0, 1\} \quad \forall i \in N, \forall k \in K, \forall t \in T \tag{3.10}$$

The network flow problem aims to minimize the distance and costs, while upkeeping the functionality of the network (3.1). The inventory level of depots is based on the previous value, demand and returns rates as well as relocations of inventory within the network. (3.2) shows the inventory definition of working stock at regular depots, while (3.3) shows the definition of broken stock at these same depots. The difference is that there is no demand for broken inventory, as well as no relocation of broken stock from one depot to another. (3.4) and (3.5) define the level of working and broken inventory at repair depots, including outgoing relocation of working stock and incoming relocations of broken stock and the repaired items.

(3.6) and (3.7) are the minimum inventory levels, where the level of working inventory should always be higher than the safety threshold and broken inventory should be positive. The capacity of depots should ways be respected (3.8). The introduction of additional stock is limited to a maximum amount P_k, as can be seen in (3.9). While (3.10) holds the binary value for functionality.

3.2.2 Minimum vehicle calculation

The calculation of the minimum number of vehicles is a variation of the knapsack problem. In the case of inventory transported in vehicles, the items in knapsacks is replaced by items in vehicles. The aim is to fill each of the vehicles with as many items as possible, minimizing the number of vehicles required for transport. Since there is an infinite number of vehicles, it can be qualified as a multi-knapsack problem.

Just as in the network flow problem, the assumption of outsourced vehicles creates an infinite number of vehicles able for use. Therefore, any amount of flow Y_{ijk}^{t} can be transported with n_{ij}^{t} vehicles. Because the vehicles being outsourced to a transportation company, no extra costs are generated outside of the vehicles used for the network. These costs, such as mentioned in the second assumption, are fixed and do not change due to situations or other external factors.

Thirdly, the last assumption holds the capacity assumption of all vehicles involved. As the main model involves capacitated vehicles with regular measurements, all capacities are equal. Additional policies such as alternative transport methods could increase or decrease the capacity of the vehicles involved.

The Mixed-Integer Linear Programming formulation of the multi-knapsack problem

Table 2. Notations used within the multi-knapsack problem

Sets	Definition
G	The complete directed graph representing the entire system $G = (N, A)$
A	All the arcs within the system, equal to all the different routes between depots
N	The set of all nodes within the system, $N = (N_{depot}, N_{repair})$
N_{depot}	The set of all depots within the system, $N_{depot} = \{1, 2, \dots, z\}$
N_{repair}	The set of all repair locations within the system, $N_{repair} = \{z+1, z+2, \dots, m\}$
T	Set of time including the time horizon H, $T = \{1, 2, \dots, H\}$
K	The set of all commodities within the system, $K = \{K_{working}, K_{broken}\}$
$K_{working}$	The set of all working commodities within the system, $K = \{1, 2, \dots, l\}$
K_{broken}	The set of all broken commodities within the system, $K = \{l+1, l+ 2, \dots, k\}$
Parameters	
D_{ij}	Distance between pick-up depot j and drop-off depot i
C_{ij}	The costs of pick-up at depot j and drop-off at depot i. This is a combined cost that consists of transport and handling cost: $C_{total} = C_{transport} + C_{handling}$
Y_{ijk}^{t}	Flow of commodity k from depot i to depot j in period t
W_k	Commodity weights per item
C_{truck}	Capacity of the vehicles within the network
Variables	
n_{ij}^{t}	Number of vehicles to flow from depot i to depot j in period t

The mathematical formulation is described in line 3.11 – 3.12 as well as the objective function, where the n_{ij}^{t} value is the number of vehicles to be minimized.

$$min \sum_{t \in T} \sum_{(i,j) \in A} \left(\alpha * D_{ij} + \beta * C_{ij} \right) * n_{ij}^{t} \tag{3.11}$$

Minimize

Subject to

$$0 \leq \sum_{k \in K_{base}} Y_{ijk}^{t} * W_k + \sum_{k \in K_{post}} Y_{ijk}^{t} * W_k + \sum_{k \in K_{shelf}} Y_{ijk}^{t} * W_k \leq C_{truck} * n_{ij}^{t} \quad \forall i,j \in N, \forall t \in T \tag{3.12}$$

Equation (3.11) includes the objective function for the multi-knapsack problem, it aims to minimize the distance driven by vehicles for relocation, the cost for relocation and the penalty for stockouts within the system. The number of vehicles n_{ij}^{t} is determined by the capacity of vehicles and the items to be relocated including their weights. In (3.12) the distribution of flow over the vehicles is determined.

3.2.3 Variation in depot capacities

Some depots do not require all of its capacity and can be reduced in size, which lowers the rent and other operational costs. If the variation in depot capacities does not lead to additional transport costs or distance, then it can be seen as a good moderation.

Given the outcome of the initial model, the minimal required depot capacities can be calculated based on the inventory values. Two variations of the used in this research are with fixed and flexible depot capacities, given in Section 3.2.1.1 and 3.2.1.2 below. An additional variation is the exclusion of maximum depot capacities within the initial model, as elaborated in the final Subsection 3.2.1.3.

The parameters and values included in the mathematical formulation are listed below.

Table 3. Notations used within the depot capacity model

Parameters	
P_i	Price per year per m^2 at depot i, which differs between depots
I_{ik}^{t}	Inventory level of commodity k at depot i in period t
M_k	Number of items per commodity per m^2
Q_k	Configuration of stacking per commodity
Variables	
c_i	Capacity of depot i

3.2.3.1 Fixed depot capacities

Taking into account that depot capacities are fixed throughout the year and the decision for the amount of capacity is made at the start of each year, the minimal amount of capacity can be calculated using the mathematical formulation below. The amount of capacity required is based on the maximum inventory of each commodity, their configuration of stacks including the size of the stacks of inventory.

Minimize

$$min \sum_{i \in N} u_i * P_i \qquad (3.13)$$

Subject to

$$u_i = u_i * (S_r + 1) \qquad \forall i \in N, \forall t \in T \qquad (3.14)$$

$$I_{ik}^{max} = max\ I_{ik}^{t} \qquad \forall i \in N, \forall t \in T \qquad (3.15)$$

$$u_i \geq \frac{I_{ik_{shelf}}^{max}}{Q_{shelf}} * M_{shelf} + \frac{I_{ik_{post}}^{max}}{Q_{post}} * M_{post} + \frac{I_{ik_{shelf}}^{max}}{Q_{shelf}} * M_{base} \qquad \forall i \in N \qquad (3.16)$$

3.2.3.2 Flexible depot capacities

As the depot capacities are directly related to the amount of inventory at each of depots, flexible depot capacities could potentially decrease the costs for rent throughout the year. Although shorter periods are better fitted for lowering the capacity required for the system to function effectively, constant changes to the depot capacities might be infeasible for implementation.

Minimize

$$min \sum_{t \in T} \sum_{i \in N} u_i^t * P_i \qquad (3.17)$$

Subject to

$$u_i^t = u_i^t * (S_r + 1) \qquad \forall i \in N, \forall t \in T \qquad (3.18)$$

$$u_i \geq \frac{I_{ik_{shelf}}^{t}}{Q_{shelf}} * M_{shelf} + \frac{I_{ik_{post}}^{t}}{Q_{post}} * M_{post} + \frac{I_{ik_{shelf}}^{t}}{Q_{shelf}} * M_{base} \qquad \forall i \in N, \forall t \in T \qquad (3.19)$$

3.2.3.3 Variable depot capacities

Within the initial model, the maximum depot capacities are part of the model constraints with regards to the following mathematical notation.

$$\sum_{k \in K} I_{ik}^{t} \leq U_i \qquad \forall i \in N, \forall t \in T \quad (3.20)$$

Removing the capacity constraints from the initial model, could result in a reduction of the total relocation of inventory by the RTI pooler. The aim of the depot capacity optimization is therefore expanded from solely minimizing the amount of m^2 of storage within the network, to using additional storage to reduce the operational costs as well as reducing the transport distance of vehicles within the system. By using the big M technique, the depot capacities are increased, and the capacity constraints are adapted.

$$\sum_{k \in K} I_{ik}^{t} \leq U_i * M \qquad \forall i \in N, \forall t \in T \quad (3.21)$$

3.3 Policies

Based on the goals of the RTI pooler, policies can be implemented to achieve the objectives of RTI pooler networks. The effects of these policies can be measured by calculating the difference in outcomes between the model with and without the addition of certain policies.

3.3.1 Depot locations

The depots are the most significant elements of an RTI pooler network, of which the number of depots, their location as well as their size are the most important aspects. Since the size of the depots is optimized in Section 5.1 and the depot locations as well as the number of depots are not fixed, both will be elaborated in this Section.

The optimization of depot locations will be performed using k means clustering, using the centers of these clusters as optimal depot locations using their weighted center of gravity. By using k means clustering, all customers within the network will be divided into k clusters based on their geographical location. Each of these clusters has a center which can either be based on solely the average location, or by using a weighted centering method that considers the size of customers while calculating the center of each cluster. In the case of RTI pooler networks, the size of customers is determined by the sum of their demand of products within a certain period. The number of clusters is free to be chosen, depending on the number of depots that is preferred within the network.

3.3.2 Purchasing decisions of new inventory

Acquiring new inventory is part of the yearly cycle of RTI poolers, to be able to keep up with the demand of customers. The trend of growing demand combined with the outflow of unrepairable items requires the amount of stock to be acquired to grow every year. The advantage of purchasing new stock is that they can be delivered to depots where they are needed the most, where new inventory can be divided over all depots in any period of time and therefore optimized within the model. The mathematical formulation of the purchasing decisions for new inventory is given in (23) and (24) below.

$$I_{ik}^{t} = I_{ik}^{t-1} + R_{ik}^{t} - D_{ik}^{t} + Q_{ik}^{t} + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^{t} + A_{ik}^{t} \quad \forall i \in N_{depot}, \forall k \in K_{working}, \forall t \in T \quad (3.22)$$

$$\sum_{i \in N} \sum_{k \in K} \sum_{t \in T} A_{ik}^{t} \leq P \quad (3.23)$$

The inventory to be purchased can be any of the three working commodities and can be delivered at any of the depots within the system, at any of the periods.

3.3.3 Repair at location

The repairs of inventory are mostly bound to repair locations within the network but are crucial for the entire RTI pooler network. In the current situation, to be repaired inventory will need to be transported to one of the repair locations to be able to keep repairing inventory. At these repair locations, substantial amounts of inventory are repaired, which does require high amounts of transport to be initiated. The addition of extra repairs at other depots than the repair locations, could therefore decrease the amount of transport between depots in the network.

$$I_{ik}^{t} = I_{ik}^{t-1} + R_{ik}^{t} - D_{ik}^{t} + Q_{ik}^{t} + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^{t} \qquad \forall i \in N_{depot}, \forall k \in K_{working}, \forall t \in T \quad (3.24)$$

$$I_{ik}^{t} = I_{ik}^{t-1} + R_{ik}^{t} - Q_{ik}^{t} + \sum_{j \in N} y_{jik}^{t-1} - \sum_{j \in N} y_{ijk}^{t} \qquad \forall i \in N_{depot}, \forall k \in K_{broken}, \forall t \in T \quad (3.25)$$

As can be seen in (3.24) and (3.25), inventory is already being repaired at depots other than the repair locations. As these amounts are relatively low compared to the amount of inventory repaired at repair locations, these repair values at depots can be increased.

Due to the strongly rising prices of resources, repairing is even more crucial then it was before. Increasing the repair rates where possible is therefore extremely cost-efficient and very interesting in systems where the repair rate could potentially be increased.

3.3.4 Additional policies

Next to the policies that can applied to the RTI pooler network and its reverse supply chain, additional policies can be applied to elements outside the scope of this research. Policies that might be fruitful but lie outside the scope of this research are policies such as alternative transport methods and price incentives, which are elaborated shortly in the sections below.

3.3.4.1 Alternative transport methods

The relocation process involves transportation, either through corporately owned vehicles or through external transportation companies. Replacing these capacitated vehicles with alternative modes of transport highly affects the sustainability and cost-efficiency of the system. Alternative transport methods such as electric vehicles, trains and vehicles with smaller capacity can be implemented within the network to improve the performance of the network. By doing so, the values of α, β and γ within the objective function varies per application. Implementing electric vehicles would reduce the need to minimize the total distance driven, thus lowering the α value and making the other two elements more important in the model: cost-efficiency and functionality. Alternative vehicles would change the multi-knapsack problem as the capacities of vehicles could be affected.

3.3.4.2 Price incentives

Additionally, price incentives have generally been shown to affect consumer behavior (Simon et al, 2010). Within RTI pooler networks, price incentives can be implemented to affects the customer's demand and returns. Through these price incentives, customers can be triggered to change the timing of their demand and returns as well as the preferred location of their demand and returns. In the resulting system, the difference between demand and returns in certain depots will be smaller and therefore less relocation by the RTI pooler.

Concluding, the inventory management of RTI poolers can be modeled using a network flow problem. The mathematical formulation as described in this thesis creates the opportunity to apply policies to the system and its network. Policies such as depot locations, purchasing decisions and repair at location can be formulated mathematically and therefore added to the network flow model. Complementing the network flow model with a multi-knapsack problem results in the last step of the scheduling, regarding the number of vehicles necessary for transport.

4. Solving the model

Methods to address the network flow problem and multi-knapsack problem with additional policies

4.1 Method

In the course of this thesis, multiple methods are used. As they complement each other, they will be used in a predetermined order and elaborated accordingly. With the help of both scientific literature as well as data from the company Container Centralen, the characteristics, elements and processes of the RTI pooling system in the case study will be researched. System elements such as the location of depots, total amount of RTIs within the network and the operational cost values are drawn from the data of Container Centralen, as well as the processes such as the repair of broken stock. Next to the case specific data, scientific literature will provide us with the theory and research on cost-effective and sustainable RTI management.

Ideally, the network of the case study resembles the findings within literature. But as the difference between theory and practice can be quite substantial, potentially tradeoffs between objectives will need to be made before moving forward with the research.

4.1.1 Mathematical programming: Linear model

Mathematical programming is the area of mathematics focused on solving optimization problems. Optimization problems are aimed at finding the minimum or maximum of a given objective function, including a set of constraints and / or conditions that solutions, consisting of a set of variables, must adhere to. Constraints set in mathematical modelling can either be equality or inequality constraints, which resemble elements of the actual problem. For the application of mathematical programming to real life business cases, the mathematical model represents the situation of the company including their KPI's in the objective function and processes in the constraints.

The data from the case study of Container Centralen and the scientific literature will be merged in a linear model, which is used to optimize the inventory management of the system. Model parameters, variables and constraint that will be crucial for the linear model are obtained by studying the given data. Combined, these elements will form the model resembling the system of the RTI pooler. As seen in research from Karsten (2015) and Lau et al. (2000), linear modeling for network flow models has been done numerous times before. The novelty of the linear model is in the combination of a network flow problem with a multi-knapsack problem, to which policies are applied. This combination within an algorithmic framework is what makes the linear model innovative.

The linear modeling using CPLEX will be performed to solve and optimize the network flow problem at hand, while keeping the KPIs in mind. For the given period of time, the linear model will propose routes, times and quantities to deliver, in order to optimize the inventory management of the RTI pooler.

As the goal is to improve the sustainability of the system while lowering the operational costs, the ideal solution involves both these KPIs. The situation might occur where there is no single optimal solution and a tradeoff between the KPIs will need to be made.

4.1.2 Sensitivity analysis

Based on the linear model, sensitivity analysis will be performed through variating with policy values and network configurations to understand the effects of the model parameters on the output values. The optimal configuration of parameters such as the maximum depot capacities, repairs at depots and depot location will be analyzed, in order to increase both the sustainability and the cost-effectiveness of the entire system. Through the implementation of policies and objectives, the sensitivity of the mathematical model to certain parameter values is addressed.

4.1.3 Multi-period network flow formulations

Multi-period network flow models are constructed out of multiple networks linked by arcs that define the connection between different temporal states (Manfren, 2012). For depots within the network, their inventory is determined from input/output flows in the current period and inventory from the previous period. The connection between the networks over time respect the chronological order of the periods, which causes the size of the entire network to be as large as the network $\times$ the number of periods used. Each period holds the entire network, including all of the relocations and the data for that period as the input data is in real-time. Since the data is in real-time, the choice for the length of the periods is crucial for the applicability of the optimization model. The demand and returns cancel each other out in the inventory definition when chosen too large, while smaller periods massively increase the calculation time.

4.1.4 Mixed integer linear programming

When a linear problem allows for both continuous as well as integer values, it is labeled as Mixed Integer Linear Programming (MILP). Mixed Integer Linear Programming (MILP) is a form of mathematical programming known for its rigorousness, flexibility and extensive modeling capability. Therefore, it has become one of the most widely explored and used methods for various problems, including optimization problems. Due to time being involved in the model, MILP is specifically applicable because of the numerous discrete decisions involved (Floudas and Lin, 2015).

Next to the overview of classic network flow problems, Hamacher et al. (2005) also introduces multiple objective linear program (MOLP). Several special cases such as the biobjective minimum cost flow problem, similar to the objective function of this thesis.

4.1.5 Branch and cut algorithm

The state-of-the-art problem solver that is used in this thesis is CPLEX, which uses branch-and-cut search when solving mixed integer programming (MIP) models. The branch-and-cut procedure manages a search tree consisting of nodes representing a subproblem to be processed. The goal is to processes active nodes in the tree until either no more active nodes are available or some limit has been reached.

A branch is the creation of two new nodes from a parent node. Typically, a branch occurs when the bounds on a single variable are modified, with the new bounds remaining in effect for that new node and for any of its descendants. A cut is a constraint added to the model. The purpose of adding any cut is to limit the size of the solution domain for the continuous problems represented at the nodes, while not eliminating legal integer solutions. The outcome is thus to reduce the number of branches required to solve the MIP.

Finally, the branch-and-cut procedure is completed once a proof of optimality is performed. The proof of optimality is achieved when the MIP gap has been brought lower than 0.0001 (0.01%).

4.1.6 Weighted sum multi-objective optimization

In order to effectively formulate an objective function based on multiple elements, the weighted sum method is applied to the objective function in this thesis. Through the use of weights, single or multiple objectives within the formulated multi-objective function are highlighted and outweigh other objectives. Which resembles the importance of every objective to the problem owner. Research by Marler & Arora (2009) has shown that the values set as weights within the multi-objective function should be chosen based on several guidelines. Weight values should be significant relative to other weights and to its objective function, objective functions should not be transformed/normalized when viewing the weights as trade-offs and it only provides a basic approximation as it is incapable of involving complex preference information. Since this thesis uses the weighted sum method in a multi-objective function for trade-offs, adhering to the guidelines will result in well-chosen weight values for all objectives within the multi-objective function. A basic mathematical formulation of the weighted sum multi-objective function is given in 4.1 and 4.2 below, using the three weights α, β and γ.

$$Y = \alpha * F_1 + \beta * F_2 + \gamma * F_3 \tag{4.1}$$

Where

$$\alpha + \beta + \gamma = 1 \tag{4.2}$$

4.1.7 Master-slave algorithm framework

Based on the work of Lau et al. (2000), the following framework for solving inventory management problems is created. The input values such as the network configuration, inventory values, demand and returns and vehicle configuration are given, in order to optimize the network configuration of the RTI pooler, the relocation scheduling by the RTI pooler and the final transportation to be initiated by the RTI pooler with the operational transport schedule. The network configuration phase will be affected by certain policies, which results in a different network infrastructure and depot configuration which are given as input for the relocation scheduling phase of the model. Which is the main optimization of the inventory management by RTI pooler performed in the master algorithm. Once this phase is completed the relocation schedule is given to the transportation phase that optimizes the use of vehicles through a multi-knapsack problem as the slave algorithm.

Lau et al. (2000) have proposed a novel algorithmic framework approach to solve a scheduling problem in supply chain management, which involves the use of multiple problems in a stepwise manner. In their study, an inventory management problem and a vehicle routing problem with time windows are solved using a Master-Slave fashioned framework. In later research by Lau et al. (2002), this framework has been applied to another inventory routing problem, using a network flow problem as the Master and a vehicle routing problem as the Slave. This research is similar to the latter research, as the algorithmic Master-Slave fashioned framework from Lau et al. (2002) has been applied to the inventory management problem with a network flow problem as the Master and a multi-knapsack problem for capacitated vehicle as the underlying algorithm.

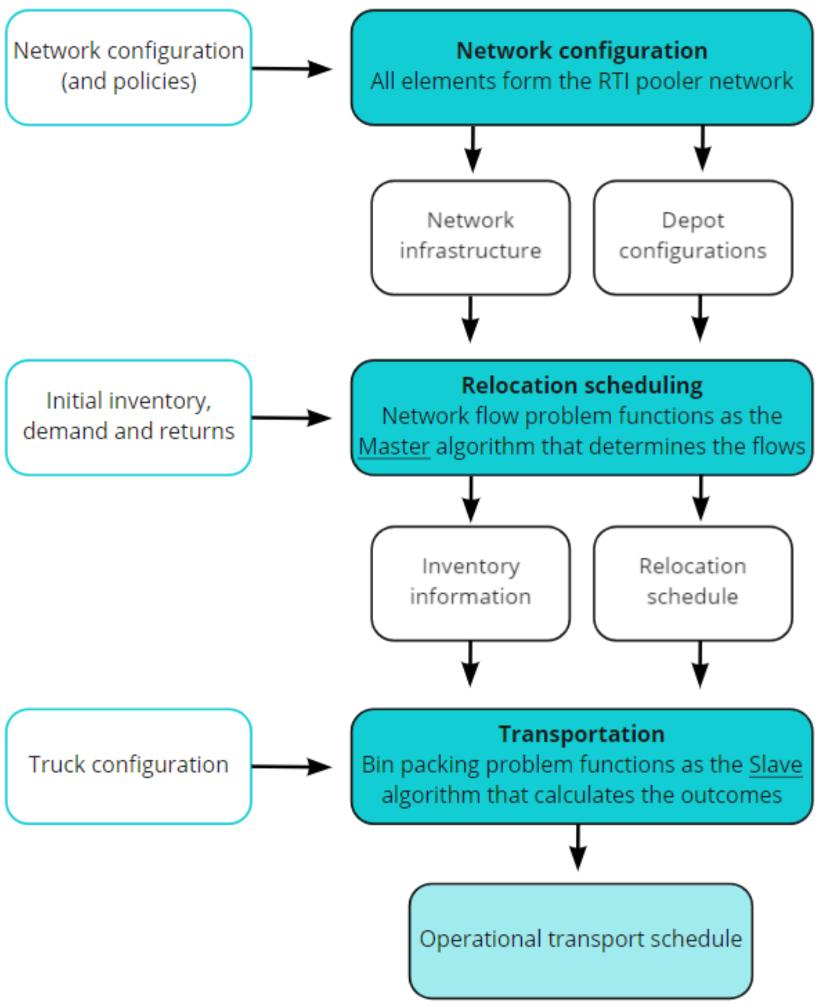

Figure 7. RTI pooler inventory management phases

4.1.8 Trade-offs between objectives

Within the service industry, both economic revenue as well as maximizing functionality are two key objectives for businesses (Dorner et al, 2011). But with the increasing concerns on environmental issues, businesses are increasingly confronted with playing a more active role to reduce the environmental burden by increasing their sustainability (Figge et al., 2012). Because of this, businesses are incorporating sustainability as an objective, which causes them to face a highly complex trade-off between three objectives: maximizing revenue, customer satisfaction and sustainability.

For the specific case of RTI poolers, maximizing revenue consists mostly of reducing their operational costs. Next to that, maximizing the functionality of their system causes the number of stock-outs to drop, resulting in the desired increase in customer satisfaction. This leads the objectives of RTI poolers to be minimizing operational costs, maximizing functionality of the system and maximizing the sustainability. The task of RTI poolers is to find a balance in the trade-off between their objectives, as can be seen in the analytical framework of Hahn et al (2010). Their framework elaborates the trade-offs to be made on societal, industry, organizational and individual level for the outcome, temporal and process dimension. The outcome dimension of RTI poolers, active in the organizational level, contains the trade-offs between different economic, environmental and organizational outcomes. While the underlying process and temporal trade-offs between different strategies and short-term versus long-term orientation highly affect the outcome dimension. As the trade-offs in all three dimensions contribute to the final trade-off between objectives for the company, they will be addressed in this thesis through the weighted sum method elaborated in 4.1.5.

4.2 Objectives

Within the mathematical formulation of the dynamic network flow problem, the objective function captures the importance of each of the objectives within the problem: sustainability, costs and functionality of the network. Each of these values is weighted in the model by the α, β and γ values, corresponding to the importance of that elements for the entire problem. Different configurations are therefore used as input for the model, which are elaborated below.

4.2.1 Goals

RTI pooling management consists of combining forward supply chain logistics with reverse supply chain logistics, including their supply chain management operations. Asedeko (2002) has researched supply chain management (SCM) and proposed a model including the goals of SCM: customer satisfaction, value, profitability and competitive advantage. Due to the sustainable nature of RTI pooler networks as well as the burden of reverse logistics it takes away from producers, the value of their supply chain is clear. However, the customer satisfaction and competitive advantage depend highly on the sustainability and functionality of the RTI pooler network. Although RTI pooling networks are sustainable in nature, effective supply chain management in the shape of forecasting and logistics can increase the sustainability and functionality of the system. And therefore, increase the customer satisfaction and competitive advantage. Finally, the profitability of the supply chain depends on both the pricing and the costs within the RTI pooler network.

Since the objective function holds the α, β and γ values, corresponding to sustainability, operational costs and functionality, the objective function can be adapted to the goals of the RTI pooler. The trade-off between the objectives within the objective function is very sensitive, as can be seen in the analytical framework for trade-offs in corporate sustainability by Hahn et al. (2010). Various decisions on both the short-term and long-term, both by smaller players and governance, influence the outcome of the model. The outcomes of the trade-off on the industry level and the organizational level ae the most crucial in the case of managing RTI pooler networks.

Based on the RTI pooler network supply chain objectives and the trade-offs between them, the following goals have been identified. Each of the goals for RTI pooler networks has an effect on each of the three objectives: sustainability, cost-efficiency and functionality. The goals mentioned below are meant to highlight one or more objectives within the objective function.

4.2.1.1 Sustainability

The sustainability of the networks' system is directly linked to the amount of distance travelled within the network. Distance is related to a fixed emission per unit of distance, leading to an amount of emission based on the total distance covered. Lowering the distance results in lower emissions by transport, therefore increasing the sustainability of the network. Within the mathematical formulation, the α value will be set relatively high, in order to minimize the distance travelled within the network. The other values β and γ will be set significantly lower in order to optimize the sustainability as best as possible.

4.2.1.2 Cost-efficiency

In a rare situation where the goal of sustainability and functionality are unimportant for the company, what remains is the goal of the lowest costs possible for the company. Although this scenario is highly unlikely, it is interesting to compare the characteristics and costs of the outcoming schedule. These costs are built-up out of out of the handling costs and transport costs, resulting in a focus purely on minimizing these two aspects while setting the α and γ values close to zero within the objective function. At the cost of distance and functionality, the cost-efficiency of the system is to be minimized.

4.2.1.3 Functionality

Preventing any stock-outs is useful for maximizing the functionality of the system. This means that there will be a built-in buffer above the safety stock to ensure the inventory level will rarely drop below that value. This can only be achieved through additional relocation on top of the relocations necessary without the buffer, resulting in additional distance as well as additional costs for upkeeping the entire network. Respecting the safety stock as well as the additional buffer at all time can be seen as the optimal way of keeping the networks' system operational and thus optimizing the functionality of the network.

4.2.1.4 Business orientated

To ensure the network stays functional at the lowest possible costs, the costs as well as the functionality should be the main part of the objective function. In this case, a schedule will be created that relocates a minimum number of items in the cheapest way possible. Distance driven, and therefore sustainability, is not important in this case as that does not directly lead to additional costs. It is generally seen as an extra operational cost, as businesses have only recently been implemented sustainability in their business objectives (Figge et al, 2012). Within the model, the distance becomes irrelevant and the focus is purely on the cost and functionality. These values will be minimized at the cost of the distance travelled within the network.

4.2.1.5 Overall minimization

When RTI poolers do not have priorities in their objectives, all of the elements within the optimization function are treated equally, since there is no element within the objective function more important than the others.

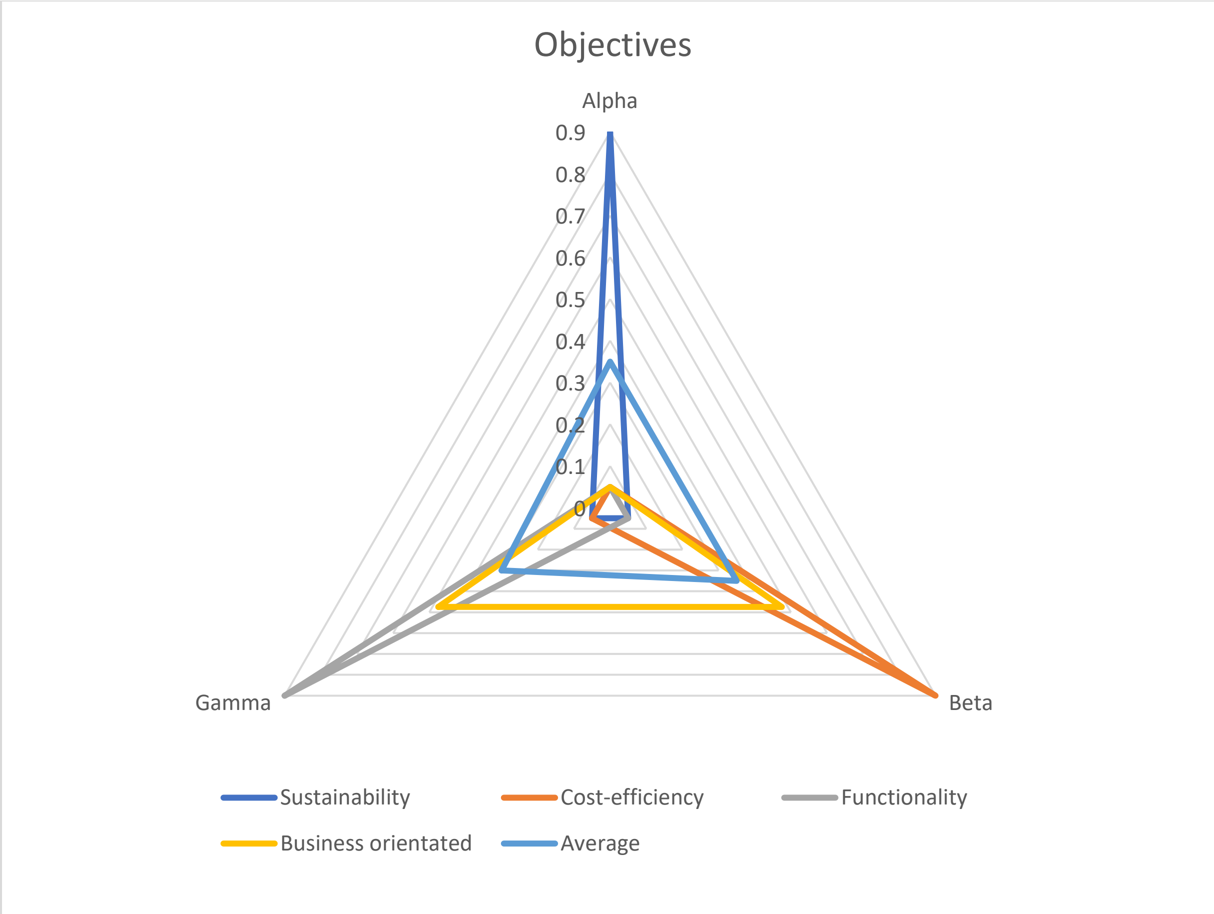

Figure 8. Alpha, Beta and Gamma values per goal

Based on the five goals described in Section 4.2.1, the Alpha, Beta and Gamma values per goal are given in Figure 8 and Table 4. As mentioned in Section 4.1.6, the values are based on the weighted sum method for multi-objective optimization. Values set as weights are chosen based on the guidelines of Marler & Arora (2009), as described in Section 4.1.6.

Table 4. Alpha, Beta and Gamma values per goal

	Objective values		
Goals	α - Alpha	β - Beta	γ - Gamma
Sustainability	0.9	0.05	0.05
Cost-efficiency	0.05	0.9	0.05
Functionality	0.05	0.05	0.9
Business orientated	0.05	0.475	0.475
Average	0.333	0.333	0.333

4.2.3 Conflicting objectives

Looking at the goals of RTI poolers, these can be divided into the category of sustainability and financial goals. Although the goals mentioned in Section 4.2.1 also include an overall objective which uses an averaged weights, the choice is either made for a sustainability focused goal or a financially focused goal which results in a conflict between RTI poolers sustainability and their financial situation. In essence, the underlying question for the RTI pooler is how much sustainability is worth in their opinion. The price of sustainability is highly dependent on the company and their financial performance, including the importance of sustainability in general for the company (Serafeim, 2018). Although the costs for the RTI pooler might increase in order to make the company more sustainable, the increased sustainability could make the company more competitive and therefore increase in size.

Regarding the objective values per goal as seen in Table 4, companies are free to set their weights differently according to their preferred values according to the trade-off between reducing costs and increasing sustainability.

5. Case study

Inventory management of RTI pooler network at Container Centralen

In this Section the previous Sections are applied to a case study, including all the knowledge gathered in the literature review. The mathematical model as well as the policies described will be applied to the case of Container Centralen. The objective is to determine the minimal amount of inventory relocation within the system's network in 2020, both with and without additional policies applied to their system. This includes the amount of distance driven, the costs for the relocation and the quality of the network itself.

For the final steps of this thesis, the company Container Centralen is used in a case study. They are a large RTI pooler within the floricultural sector with depots all over Europe, containing several million RTIs combined. As they are aiming to increase their sustainability and cost-efficiency, without decreasing the functionality of their service. Their system and its' network configuration are elaborated below.

5.1 The network of Container Centralen

Container Centralen (CC) is a large RTI pooler and the owner of the CC Pool System, the largest RTI pooling system in the European and American branch for plants and flowers. Their RTI pooling system consists of over 60 depots, with coverage of most countries in Europe. The network and its' depots are responsible for distributing the 3.6 million RTI's to be picked up by over 23 thousand customers in 40 countries.

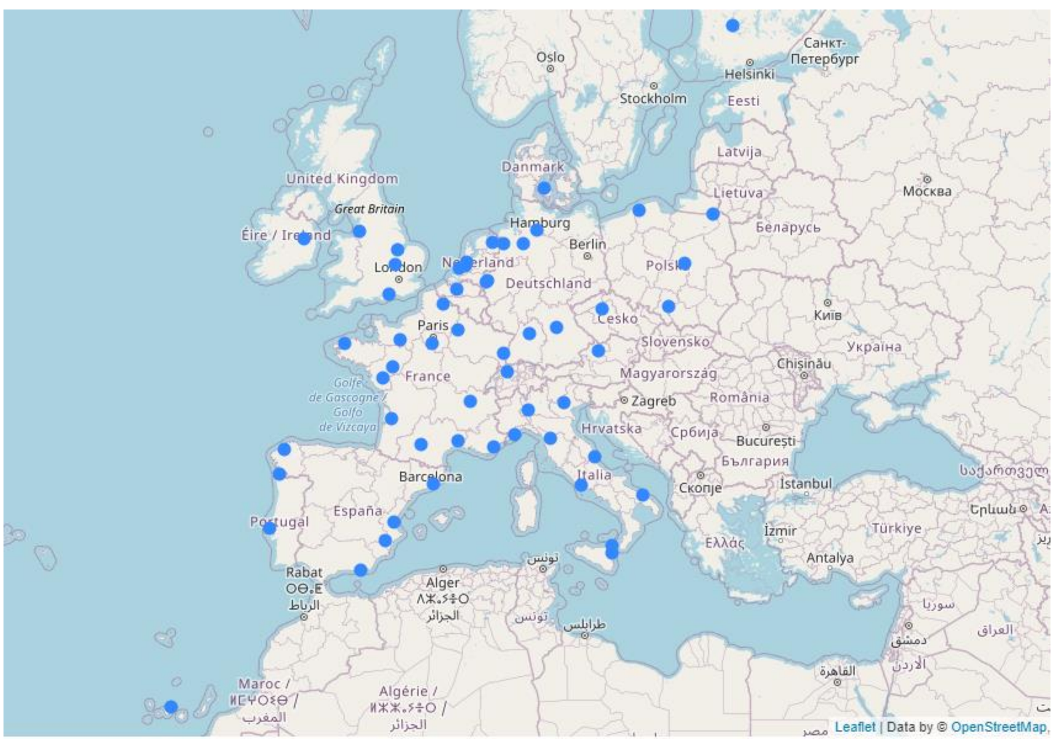

Figure 9. Depot network of Container Centralen

Table 5. Specifications of the CC RTI pooling system

	Container Centralen
Number of regular depots	56
Numbers of repair depot	4
Numbers of vehicles available	∞

The RTI's within the system are used to store items for transport as well as in storage and use by customers. For use by customers, each of the RTI's are built up out of one base, four posts and a variable number of shelves, depending on the size of the items stored. However, for actions by the RTI pooler such as relocation and storage, the RTI's are stacked more efficiently to optimally use both depot and vehicles capacities. An example of the RTI's can be seen in Figure 10 below, regarding both the state it is used in by customers on the left as well as an example of one of the stacking configurations on the right. The left shows the regular configuration for customers using a single base, four posts and four shelves in this case, where the number of shelves is dependent on the size of the items stored within. On the right you can see a stack of ten bases, as an example of how items can be stored at depots and shipped between depots. The capacity of vehicles and depots are efficiently used through the use of stacking.

Figure 10. Example of commodity of Container Centralen

5.1.1 Historical relocation data analysis

In order to compare the results of the mathematical model, the relocations between depots within the system in 2020 are analyzed. Specifically, all values for each of the three commodities. The demand and returns values within the network are based on the actual data of 2020, which results in deterministic demand and return values within the model.

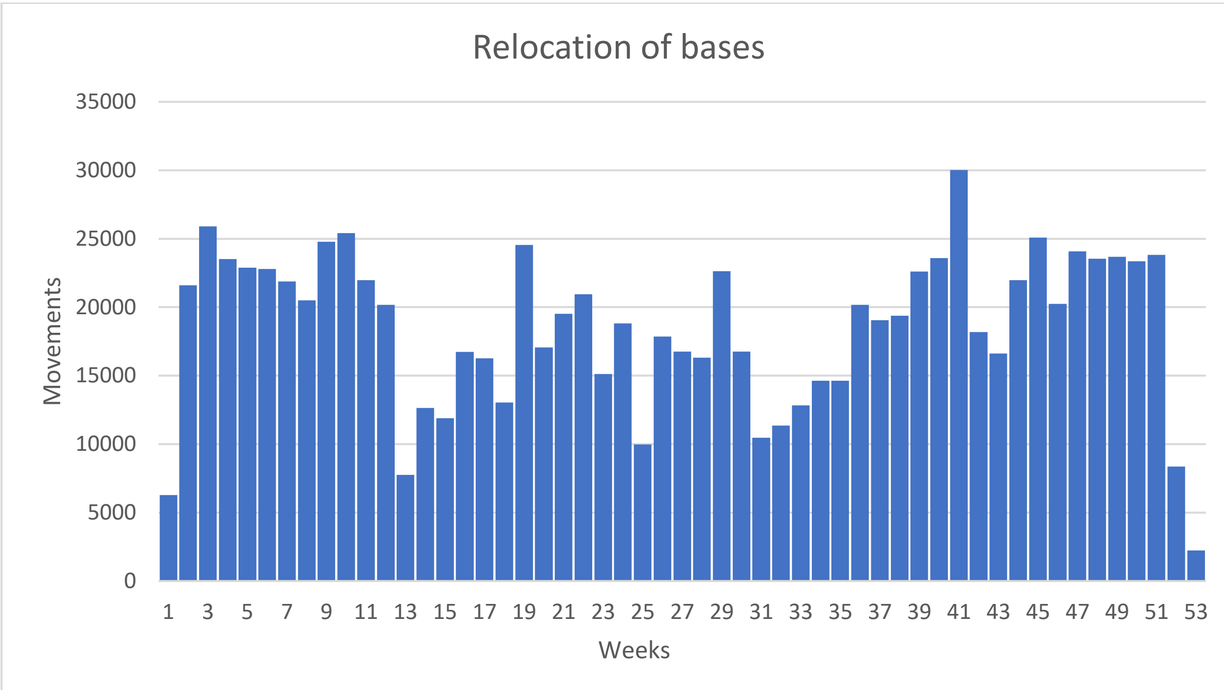

Figure 11. Relocations in the RTI pooler system over time

Summing the relocation values over the entire year, the number of relocated items is given in Table 6 below.

Table 6. Historical relocation values

	Number of relocated items		
Commodities	Bases	Posts	Shelves
Total	983.507	2.450.017	4.945.361

5.1.2 Demand and returns

Based on the demand and returns of inventory within the system in 2020, relocations between depots are initiated. Binning the data into weeks for 2020, the demand and return values are shown in Figure 12 below. As can be seen, the demand of functional stock is higher than the returns of functional stock, resulting in the need for relocation in order to make sure all depots have enough inventory available.

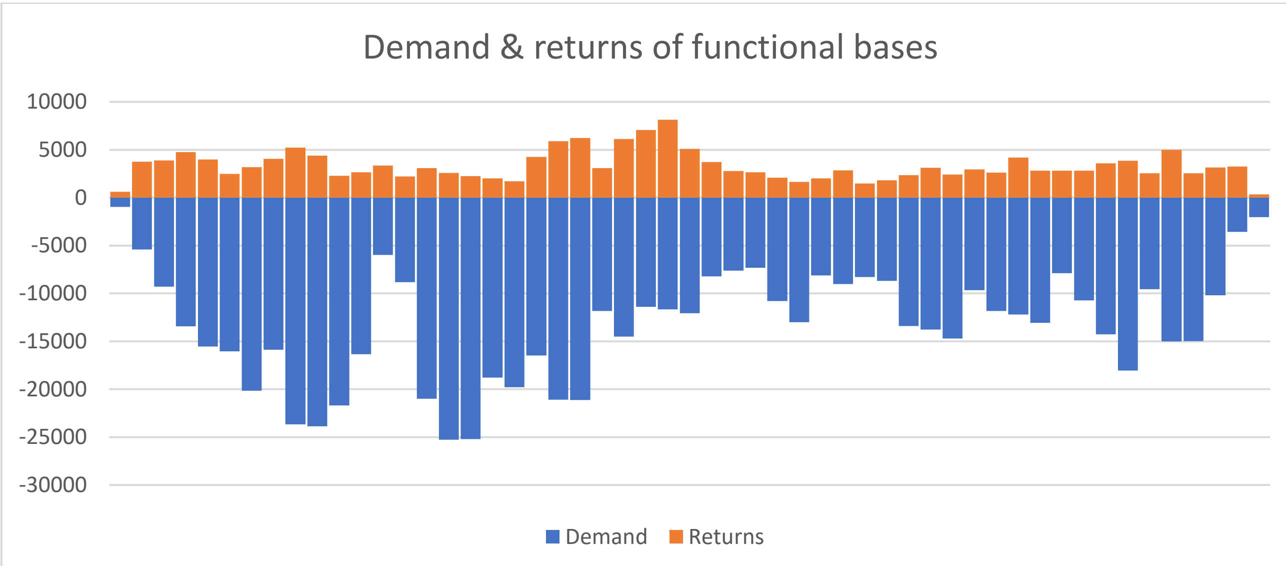

Figure 12. Demand and returns over time

The difference between the demand and return values are relatively large, as can be seen for functional bases in Figure 12. This is partially due to the exchanges of broken inventory for functional inventory by customers at depots, as these are not counted as returns of functional bases. Including these exchanges and thus reviewing the entire demand and returns of all bases within the system, results in the values shown in Figure 13.

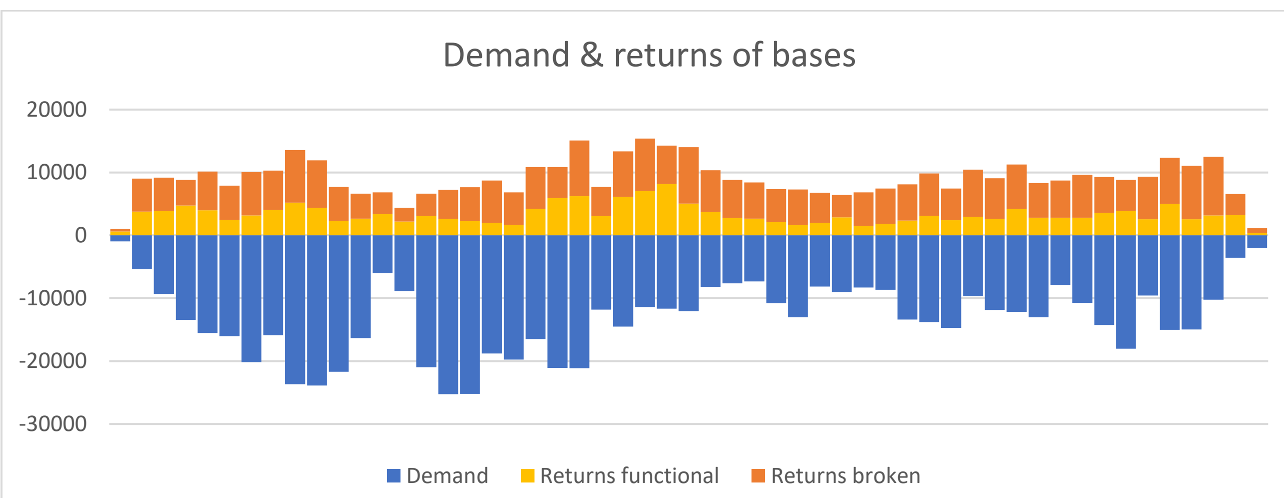

Figure 13. Demand and returns of bases over time including broken items

The difference between the demand and all returns of bases within the network in Figure 13 is relatively small due to the inclusion of returns of broken material. Due to the repair locations within the system, these broken items can be repaired and reintroduced into the system to fulfill the demand of customers.

5.2 RTI pooler inventory management optimization

Optimization of the relocation schedule within the RTI pooler network without any additional policies is performed through the master network flow model and the slave multi-knapsack problem given in Section 4.2 and 4.3. The results of the optimization models will be compared to the initial model values as described in table 5 above.

Table 7. Results per objective function

	Relocation values			Output values		
Objectives	Bases	Posts	Shelves	Distance [km]	Costs [€]	Stock out [€]
Sustainability	473.084	983.146	2.065.994	739.484	2.382.189	30.000.000
Cost-efficiency	832.975	2.263.768	4.169.247	1.000.612	1.833.697	29.000.000
Functionality	738.351	2.022.498	3.941.981	910.410	1.800.707	0
Business orientated	835.205	2.264.678	4.158.224	1.006.132	1.830.446	0
Overall	738.351	2.022.498	3.941.981	906.109	1.793.799	0

As the overall optimization policy is relatively best for all KPI's involved for the RTI pooler network, the overall minimization policy values have been used for the results in this case study.

Table 8. Optimized flow relocation values

	Number of relocated items		
Commodities	Bases	Posts	Shelves
Nonbroken	504.316	1.430.961	2.886.122
Broken	234.035	591.537	1.055.859
Total	738.351	2.022.498	3.941.981

Table 7 and Figure 14 indicate the results of the master algorithm regarding the RTI pooler inventory management optimization. The overall number of items to be relocated according to the schedule given by the network flow problem is built up out of nonbroken and broken inventory, as shown in Table 8. Comparing the results of Table 8 with the standard relocation numbers in Table 6 quickly shows that the optimized network flow model does generate a better solution regarding the items to be relocated. How these relocations are divided can best be seen on a more detailed level that indicates the flow for every route over time, as seen in Appendix B.

On an aggregated level, the sum of flows for every period in time indicate the number of items to be relocated which can be compared with the current amount of relocations in the RTI pooler network of Container Centralen. This comparison can be seen in Figure 14, which shows the substantial difference between the current situation and the optimized solution.

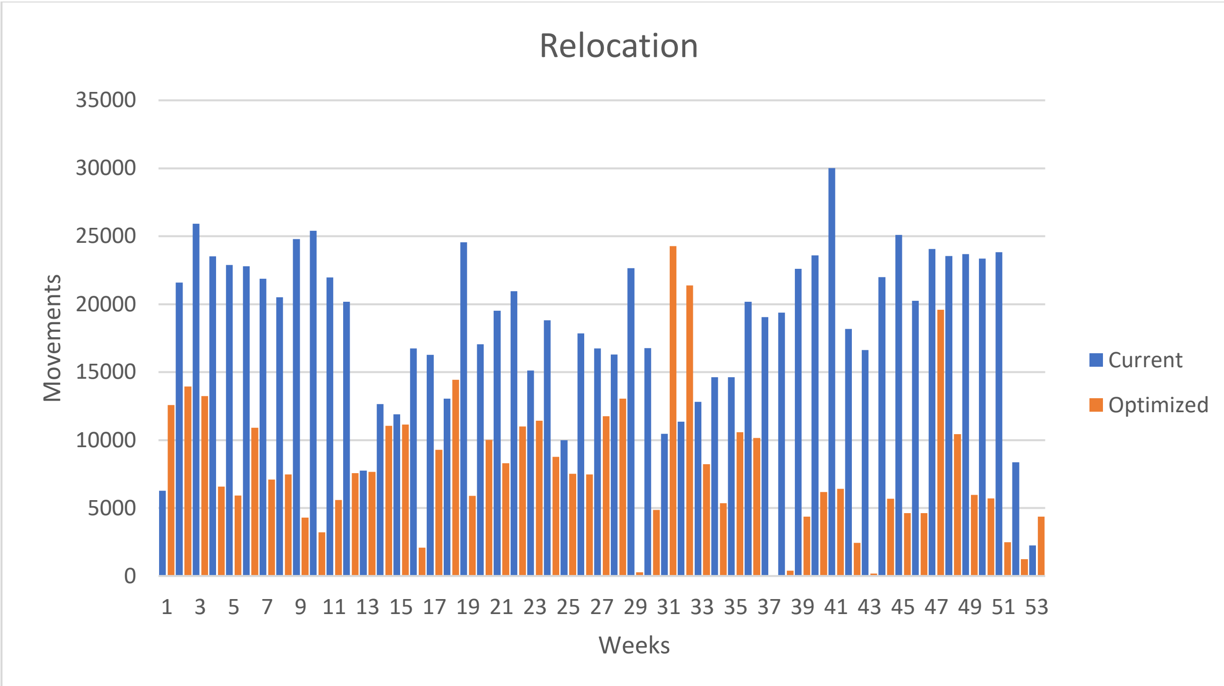

Figure 14. Optimized relocation schedule for overall minimization

Based on the different objectives set for the RTI pooler, the result does present a more efficient relocation schedule for its network. The chosen objective does however have a large impact on the relocation schedule and therefore the entire outcome of the model. Setting the correct objective for the RTI pooler within their objective function is thus crucial for the outcome of the model.

Model run statistics

The master algorithm of the RTI pooler inventory management optimization found an optimal solution within 250 seconds, for any of the objectives described in Section 4.2 and Section 5.2.

5.3 Policy implementations

The effect of the application of policies to the case of Container Centralen will be addressed separately in the Sections below. As the policies are applied to the calculated optimal relocation schedule for RTI pooler networks, their effects will be compared to the optimal relocation schedule without additional policies.

5.3.1 Variation in depot capacities

The current depot capacities with the corresponding rent are high costs for Container Centralen, which can be changed relatively easy. The depot capacities are directly related to the amount of inventory stored at each depot, more specifically the maximum number of items stored at the depots.

5.3.1.1 Fixed depot capacities

When setting a fixed depot capacity which is calculated at the beginning of each year, the rent costs could be lowered if forecasts of the inventory for that specific year show that capacity will not be used. Taking into account the current depot capacities and their corresponding rent costs, a 22% reduction of annual rent cost can be realized.

Table 9. Fixed depot capacities rent costs

Depot capacity	Depot rent [€/m^2]
Not optimized	5.297.154
Fixed depot capacities	4.137.026
Difference	1.160.128 (22%)

5.3.1.2 Flexible depot capacities

Additionally, depot capacities could also be rented with a flexible contract. Depending on the length of the periods, capacities could therefore be as low as possible when applied to the case of Container Centralen. Using the mathematical flexible depot capacities formulation as described in Section 3.2.3, it can be seen as a pay-per-use method for the flexible depot capacities. The resulting forecasted depot costs for the entire year is therefore lower than using fixed depot capacities as they are currently, since none of the rented capacity is paid for unnecessarily.

Table 10. Flexible depot capacities rent costs

Depot capacity	Depot rent [€/m^2]
Not optimized	5.297.154
Flexible depot capacities	3.721.684
Difference	1.575.470 (30%)

For both the fixed and the flexible depot capacity model, the outcome shows that there is room for improvement regarding the size of depots. Especially when a flexible depot capacity can be implemented, the rent costs for the depots can be reduced.

Model run statistics

Both of the slave depot capacity optimization models found an optimal solution within 2 seconds, as the mathematical model is relatively small.

5.3.1.3 Variable depot capacities

Within the master network flow problem, the capacities constraints of depots within the network could lead to additional relocation to be initiated. Through increasing the maximum capacity at each depot M times, the depot capacity will lead to a lower number of relocations. Looking at the outcome of the main optimization model without capacity constraints, the results indicate that relocations can be prevented through increasing the depot capacities. It is therefore an interesting policy to possibly implement to the network of Container Centralen.

Table 11. Variable depot capacities in overall minimization

	Relocation values			Output values		
Depot capacity	Bases	Posts	Shelves	Distance [km]	Costs [€]	Stock out [€]
Not optimized	738.351	2.022.498	3.941.981	906.109	1.793.799	0
No capacity constraints	450.489	1.181.747	2.264.849	597.486	1.138.564	0

Model run statistics

As the model with variable depot capacities requires a re-run of the master algorithm, it takes the optimizer longer to find an optimal solution, more specifically 170 seconds.

5.3.2 Depot locations

For the network of Container Centralen's RTI pooler system, the number of depots as well as their location are crucial. The depot locations are determined by multiple factors such as the demand, geographical coverage and pricing for land and labor. While the numbers of depots are dependent on the number of customers and the configuration of the network that already exists. New depots are likely to be created on the edges of the RTI pooler network, which increases the reach of the system. Other depots can also be replaced within a country do have a better distribution over the geographical area.

Basing the depots locations on the locations of customers using k means clustering and weighted centers such as explained in Section 3.3.1, the depot network of Container Centralen would be as can be seen in Figure 15. Where the red dots resemble the old depot locations and the blue dots the new proposed locations.

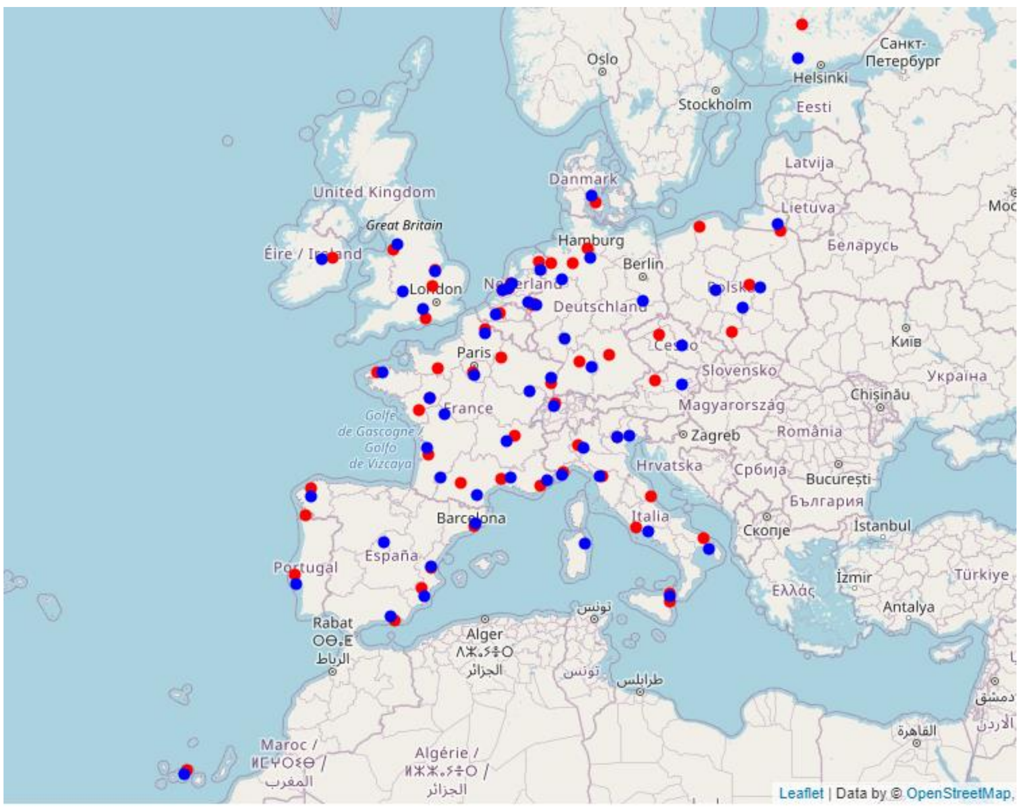

Figure 15. New and old depot locations for the RTI pooler network

By looking at the example of Spain within the entire RTI pooler network of Container Centralen, a more detailed view of the k means clustering method with weighted center of gravity is shown in Figure 16. Based on the locations of the customers shown in blue, a new location for the six existing depots are calculated. Old depot locations are shown in red, while proposed new depot locations are indicated by the green circles.

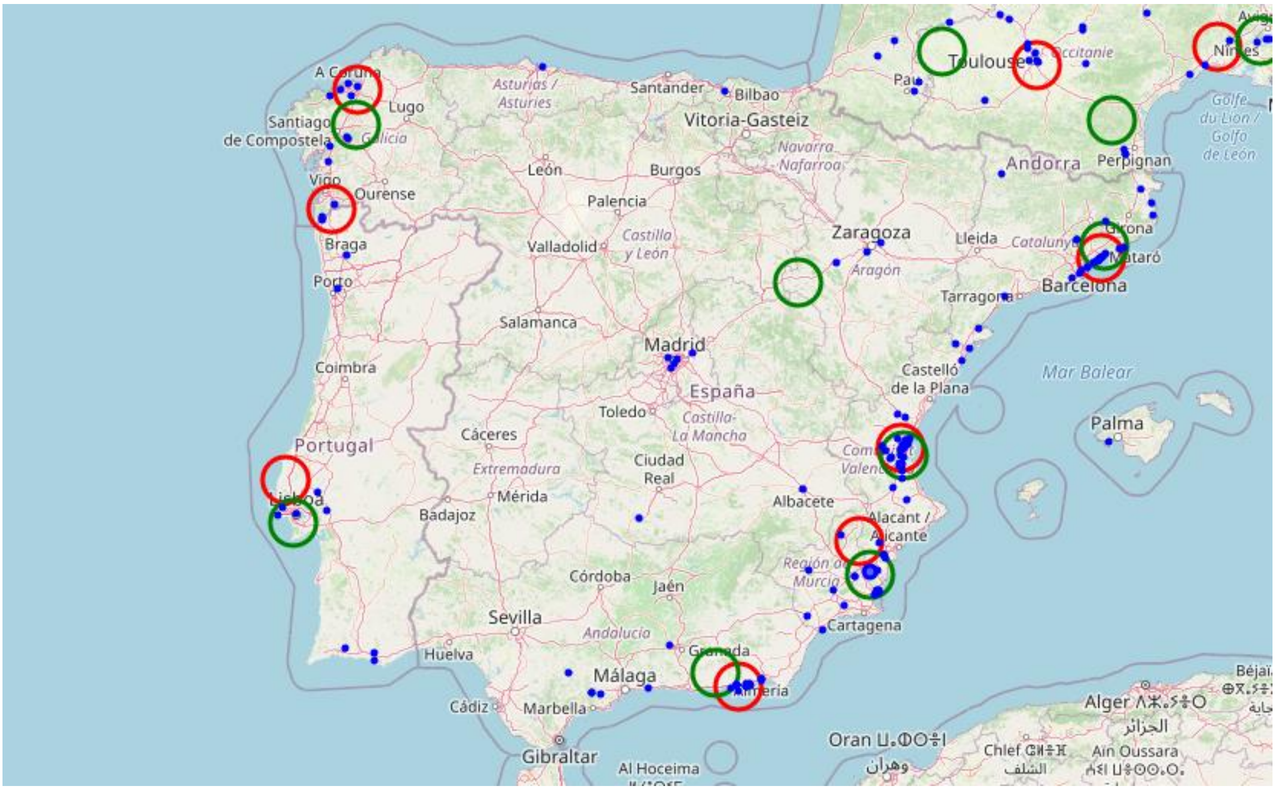

Figure 16. Old and new depot locations with customers

The results indicate that for most of the depots, the optimal locations based on the customers are relatively close to the current depot locations. However, the results also indicate that the location of two of the current depots are not optimal and could benefit from altering the depot location. As there might be multiple reasons for placing the depot in that specific location in the first place, the results should be looked at carefully in order to determine if the new proposed location is a viable option.

5.3.3 Purchasing decisions of new inventory

The purchasing decisions of new inventory are important for the amount of demand to be fulfilled as well as easier distribution of the inventory itself. A surplus of inventory within the system helps with the reduction of relocation necessary for the RTI pooler system to function.

In the current situation, a production order for new inventory is placed at the beginning of each year. The number of items ordered in the year 2020 by Container Centralen is given in Table 8, which is produced and delivered in one specific depot.

Table 12. Depot purchasing values

Periods	Bases	Posts	Shelves
1	0	0	0
2	20500	105000	74355
3	5000	59807	0
4	0	0	21088
5	0	0	0
6	9500	0	27350
7	5391	0	58900
8	0	0	43500
9	0	0	18000
10	0	0	50250
11	12700	72000	55000
12	12250	35000	50500
13	10050	33000	48912
Sum	75391	304807	447855
Purchased stock	75000	300000	450000

The number of ordered items for the year 2020 are included in the main mathematical model, whereas the addition of extra items at depots where they are required the most would result in the situation shown in Table 9, Figure 16 and Figure 17. As can be seen in Table 9, introducing additional new inventory to the system in the form of 75000, 300000 and 450000 items will result in a large reduction of the relocation needed to keep the system functional. Although the costs of acquiring and producing new inventory are substantial, the benefits that are to be made from this are substantial.

Table 13. Optimized network with additional inventory

	Relocation values			Output values		
Policies	Bases	Posts	Shelves	Distance [km]	Costs [€]	Stock out [€]
Optimized	738.351	2.022.498	3.941.981	906.109	1.793.799	0
Purchasing decisions	599.730	1.448.124	2.435.403	525.429	1.258.182	0

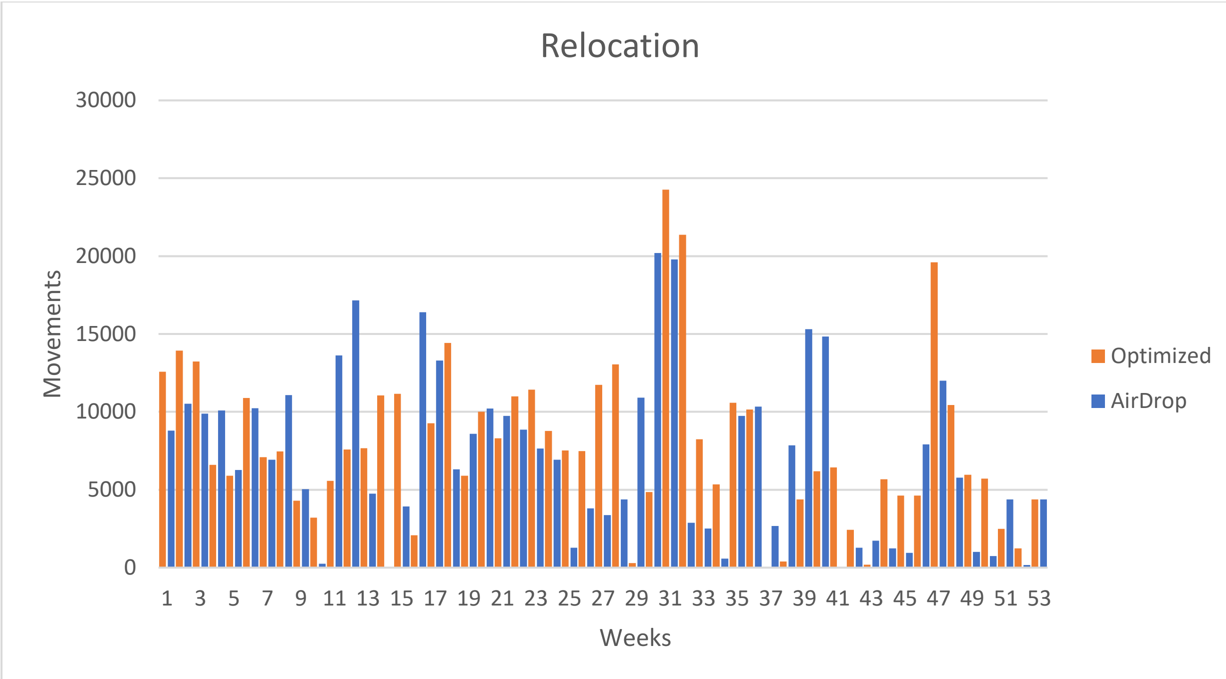

Figure 17. Relocation schedule for overall minimization with purchasing decisions of new inventory

Overall, the optimized model without adding purchasing decisions of new inventory relies solely on the relocation of stock from other depots, while including these purchasing decisions would replace some of the relocations and will therefore lower the amount of relocations necessary as expected. How the new inventory is distributed over certain depots is given in Figure 17, where can be seen that only five depots receive this new inventory while others do not require and receive any of the newly introduced items.

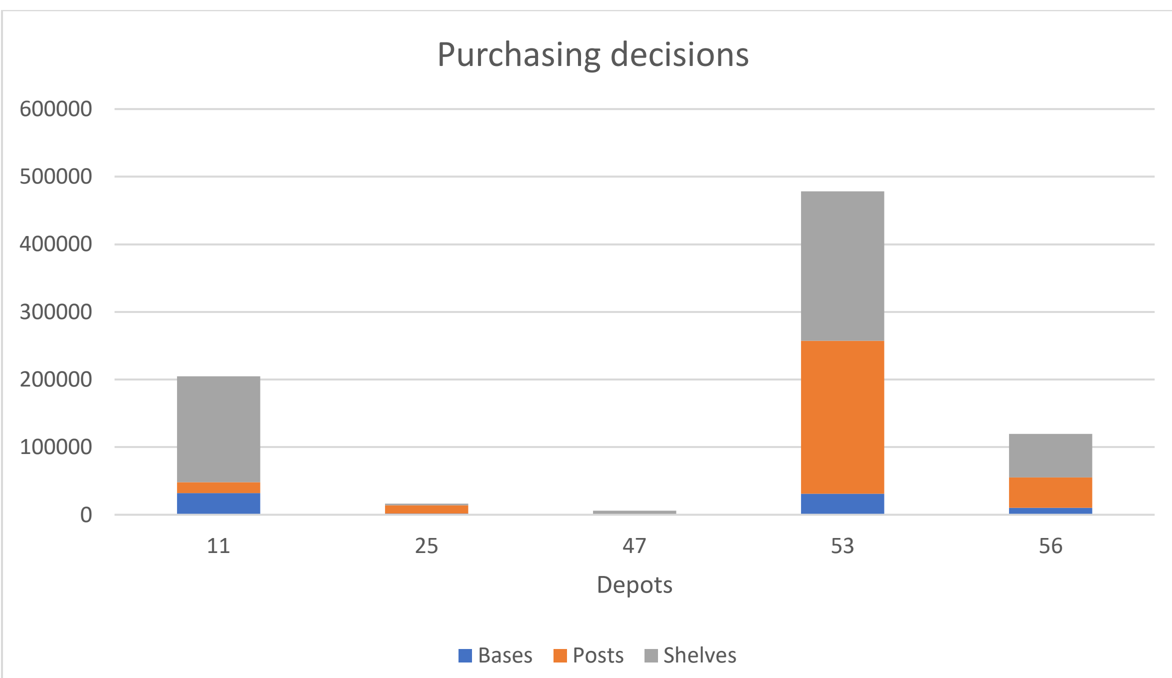

Figure 18. Purchased inventory delivered per commodity per depot

5.3.4 Repair at location

Currently, repairs occur at the specific repair depots within the network. For this to able to happen, relocation of broken stock from regular depots to repair depots needs to be initiated. Repaired items at the repair locations also need to be reintroduced to the system by relocating transporting them back to a regular depot within the network. Repairing items at regular depots within the network would therefore reduce the number of items to be relocated to repair locations while still being able to reintroduce items into the system.

The difference in the resulting system where extra repairs are added to the standard network, is given in cost efficiency, sustainability and functionality within the system. The main difference in results within this research will be found in the amount of distance traveled and the transport and handling costs associated with the relocations.

As the amount of inventory to be repaired by normal depots can be seen as an input variable, different values are used as the repair quota to calculate the effects of adding repairs at regular depots. The values used as well as the output values are given in Table 14 below.

Table 14. Repair and output values

Repair values / period			Output values		
Bases	Posts	Shelves	Distance [km]	Costs [€]	Stock out [€]
0	0	0	906.109	1.793.799	0
10	10	25	900.824	1.733.692	0
50	50	125	948.256	1.740.738	0
100	100	250	1.392.085	2.164.404	0
250	250	625	3.216.809	4.471.702	0
500	500	1250	7.653.373	12.786.698	0

Additional to repairs at all location, the option for adding an extra repair location at one of the regular depots is investigated. The depots chosen for this are depots within larger remote countries without an already existing repair location, where the largest depots will be added as a repair location as well. Some of the options for additional repair locations are given in Table 15.

Table 15. Repair and output values new repair locations

Location		Repair values / period			Output values		
Country	Depot location	Bases	Posts	Shelves	Distance [km]	Costs [€]	Stock out [€]
Current	Current	0	0	0	906.109	1.793.799	0
France	Angers	500	500	2500	903.545	1.742.887	0
		1000	1000	5000	940.704	1.783.582	0
Spain	Alicante	500	500	2500	984.750	1.923.198	0
		1000	1000	5000	1.294.200	1.315.511	0
United Kingdom	Bedford	500	500	2500	812.916	1.600.547	0
		1000	1000	5000	817.009	1.575.417	0

Reviewing the results of additional repairs at all existing depots within the system, Table 14 shows that the additional repairs at all depots does not lead to a significant reduction in relocation. Increasing the repair rate at all depots is therefore not as viable. However, selecting specific depots for creating an extra repair location instead of all locations does create an interesting opportunity for Container Centralen as seen in Table 15. Especially the addition of a new repair location at the current Bedford depot in the United Kingdom would result in a reduction in distance driven and the costs associated. The exact location as well as other elements such as the number of items to be repaired could be optimized through the process of iteration.

6. Discussion and conclusion

The goal of this research was to develop an optimized relocation schedule for the inventory management in RTI pooler networks, including the application of policies to the mathematical formulation of the network flow problem. Firstly, RTI pooler networks and inventory management in general are elaborated through a literature study. The third Section involves the mathematical formulation of the network flow problem at hand, as well as several policies to be applied to this network. Section 4 describes the way this problem is tackled, including the methods used within the rest of this thesis. The objectives of RTI poolers to optimize relocation schedule according to their preferences are also given in section 4. In the last Section 5, the mathematical models including the additional policies are applied to the case of the large RTI pooler Container Centralen in a case study. The discussion and results of this thesis will be presented in this section.

This thesis is focused on answering the research question:

" How to optimize the planning of RTI pooler networks, that seek to maximize sustainability, cost-efficiency and functionality, with respect to different policy applications to the network?"

By formulating a network flow problem in combination with a multi-knapsack problem, the planning of relocation within RTI pooler networks can be optimized effectively. Due to the objective function involved in the formulation, the relocation schedule can even be optimized based on the goals set by the RTI pooler. The different weight values are set within the objective function based on the weighted sum method, which is able to effectively determine an optimal relocation schedule for the network flow problem described in this thesis.

The addition of policies to the network flow problem in the shape of additional variables and constraints within the model has been shown to be effective as well. The application of the proposed policies is done separately but still shows promising results when applied to the case of Container Centralen. For each of the sections within this thesis, the conclusion and discussion is given in the corresponding sections below.

Inventory management optimization

In Section 3, an optimization model for the inventory management of RTI poolers was developed to generate an optimized planning of relocation within RTI pooler networks. The optimization model consists of an inventory routing model defined as a combination of a network flow problem and a multi-knapsack problem, as seen in Section 3.2. Based on a master-slave algorithm framework, the network flow problem serves as a master while the multi-knapsack problem serves as the slave which relies on the output of the master. The combination of these problems results in an optimized relocation schedule which is applicable to RTI pooler networks.

Applied to the case of Container Centralen, the resulting relocation planning is more efficient than the current planning within the chosen time horizon, meaning that the inventory management can be optimized effectively through the use of the proposed planning. As the optimized planning is able to reduce the number of relocations by up to 20% without any additional policies, the resulting planning should be studied closely to seek possible improvements for the relocation schedule of next year.

It is important to keep in mind that the optimized relocation planning is based on historical values, while the current planning is based on forecasts. The resulting planning should therefore be used for comparison, in order to see where possible improvements in the schedule can be made. In the case of Container Centralen, comparing the optimized schedule with the current schedule shows that the anticipation of possible forecasted collection is too big and can be reduced in order to reduce the number of items to be transported.

Effects of policy implementations to RTI pooler networks

Additional to the master network flow problem and the slave multi-knapsack problem, policies are implemented to the RTI pooler system in order to optimize the planning of relocation within the system. The policies to be implemented individually are either changing the network configuration of RTI poolers based on the network as a whole or for each depot within the network or affecting the network elements regarding the relocation of inventory. The policies to be implemented on RTI pooler networks can be found in Section 3.3.

All the policies mentioned in this thesis show promising results once applied to the case of Container Centralen, as can be seen in Section 5. Changing depot capacities, purchasing new inventory and additional repair locations have shown to be able to reduce the distance driven and costs by 10 to 20%. Although there might be additional costs paired with the implementation of these policies, each of them shows promise for implementation in RTI pooler networks.

Altering depot capacities is relatively straightforward to implement if the surface is being rented at an external party and shows a reduction of 20% in rent costs based on the optimized planning schedule which in itself is already more optimal than the current planning. The decisions of the depot locations is also relatively easy to implement if they are not owned by the company and could increase the functionality of the network as a whole. Purchasing decisions and additional repair locations focus on the (re)introduction of functional material into the network at certain depots, reducing the need for relocation between depots within the network. As expected, the possible reductions differ based on the number of items to be purchased or repaired, as can be seen in Table 13, 14 and 15. As these are based on the case of Container Centralen, the application of policies should be addressed carefully in order to maximize its potential. Interestingly, repairing items is equally effective at reducing the required relocation than purchasing new inventory. Meaning that the purchasing decisions should be revised and repairing should be seen as a viable alternative.

Cumulative effects have not been researched in this thesis since that complicates measuring the effects of applying policies, but could be used to optimize the relocation schedule even further than can be seen in the case study within this thesis. Each network has its own strong and weak points, so the application of policies should be addressed for each network individually to be able to optimize the scheduling for different RTI pooler networks.

RTI pooler network goals

The models mentioned in Section 3.2 and 3.3 were developed in order to determine the optimal relocation planning for RTI pooler networks, based on the goals set by the RTI poolers. These goals are based on the supply chain management objectives, as RTI poolers include both forward and reverse supply chain logistics. The goals can be categorized into sustainability goals or financial goals, resulting in a split between cutting costs and thus increasing profits or increasing the sustainability of their network.

In order to effectively apply the master optimization model to a real-life situation and use the resulting planning as a guideline, it is thus important to formulate the goal of the RTI pooler involved. Together with the goal of the RTI pooler involved, an indication should be made to what extent they are willing to sacrifice economic benefits in order to become more sustainable as a whole. As seen in Table 7 in Section 5.2, the output values of the model can differ a lot based on the goals used while modeling through the values set as weights in the objective function.

As expected, the resulting schedule is dependent on the chosen weights based on the RTI pooler goals. However, the outcome is extremely sensitive to the weights set and differ a lot because of them. Table 7 shows that the difference in distance driven and costs can differ up to 30 to 40%, which means that the weights given to the RTI pooler goals should be done carefully in order to optimize the resulting relocation schedule according to their company-specific objectives. This sensitivity does confirm the effectivity of the weighted sum method within objective functions in mathematical modeling.

Finally, looping back to the main research question.

" How to optimize the planning of RTI pooler networks, that seek to maximize sustainability, cost-efficiency and functionality, with respect to different policy applications to the network?"

The relocation planning within RTI pooler networks can be optimized through the application of the inventory management algorithm framework consisting of a network flow problem and multi-knapsack problem. Through mapping the goals and objectives of RTI poolers, the optimal objective can be found for any RTI pooling network looking to apply the model. The network configurations as well as other network elements are reliant on the policies implemented to the RTI pooler network, resulting in potentially interesting networks based on the addition of policies such as introducing additional inventory or altering depot capacities. Through the combination of the mathematical models, the goals of RTI pooler networks and the implementation of policies, an optimal solution can be found for any RTI pooling network.

The proposed models within this thesis can optimize the relocation schedule effectively, as can be seen in the case study in Section 5. In this specific case study regarding the case of Container Centralen, the optimized relocation could reduce 20% of the total relocations required within the system without affecting the functionality. While applying additional policies could realize an additional 10 to 20% reduction when applied separately and correctly.

6.1 Recommendations for further research

Multiple recommendations can be identified for future research, regarding both the mathematical model as well as the scope of the research.

Recommendations regarding the mathematical model

- **Merging the models**: Firstly, the mathematical model is two-fold since it is split into a master and a slave according to the algorithmic framework of Lau et al. (2000), while the model can be improved through combining the models and merging the algorithms into one inventory management optimization model containing both the network flow problem and the multi-knapsack problem.
- **Difference in transit times:** In the current situation, a week of transit time is taken into consideration which holds the preparation, loading, transport and unloading of items combined with additional time as the anticipation period. Since the time it takes is different for any of the depots involved and are easily split up, the model would benefit from specified transit times per arc and per depot.

Recommendations regarding the policies

- **Additional policies**: Potentially interesting policies be researched in order to effectively improve the planning of relocation within RTI pooler networks, as mentioned in Section 3.3. Due to the complexity of these policies, these have not been included in this thesis.
    - **Price incentives:** The addition of price incentives could tempt customers into changing their demand or returns. Although the exact effectivity of the price incentives is hard to measure, price incentives as a policy have been shown to be effective in other research.
    - **Alternative vehicles:** The multi-knapsack problem is highly affected by the capacity of the vehicles involved, as well as the costs and sustainability of the system. Alternative vehicles could drastically improve the sustainability and cost-efficiency of the RTI pooler network but have not been included in the model policies because of the changes in the network structure it requires.
- **Changing depot locations:** The depot locations are based on the delivery location of all customers within the network, including the size of their demand. Creating clusters using k means clustering as explained in Section 3.3, the optimal depot locations are calculated for any number of depots within the RTI pooler network. However, the demand and returns per depot are affected by the location of the depot. Changing the location of a depot will therefore alter the demand and returns per depot, which should be implemented in the model in order to achieve better results.

6.2 Reflection on research approach

Due to the novelty of applying multiple policies to a linear programming model for optimizing inventory management, I am happy and satisfied with the outcome of this thesis. It was however quite difficult to keep my focus on implementing policies at first, since the backbone of the model does not initially includes adding policies. This meant that I found myself bouncing between mathematical modeling and defining policies quite often, which took several iterations before being able to confidently apply policies to the mathematical model. Therefore, policy application could have played an even bigger role if taken into consideration earlier in the process. In that sense, the focus of my research could have been better placed on RTI pooler networks, mathematical formulation and the policies at first. Although these first sections could have been tackled differently, these are where I have learned the most about mathematical modeling, even more about applying policies and several methodologies such as weighted sum and Mixed Integer Linear Programming.

The use of the case study for Container Centralen has brought me a lot of specific knowledge on processes in the logistics sector, both on an academic as well as a on practical level. It took me quite some time to translate the physical system into a technical network as seen in literature, which I had to keep improving during the course of this thesis. My focus was therefore on improving the description of the RTI pooler network during all phases of this research, which made it difficult to separate other elements and address them individually. The focus could have been more with applying all knowledge and models to the case study once those are completely formulated, creating a clear division.

Overall I am still confident that the methods used in this thesis are applied properly and the most suited methods for defining a relocation schedule for inventory management in RTI pooler networks. By separating the process more, the research approach could have been improved.

Reference list

1. Accorsi, R., Baruffaldi, G., Manzini, R., Pini, C. (2019). Environmental Impacts of Reusable Transport Items: A Case Study of Pallet Pooling in a Retailer Supply Chain. *Sustainability, 11.* Retrieved on November 20 2020, from https://doi.org/10.3390/su11113147.
2. Accorsi R., Cascini, A., Cholette, S., Manzini, R., Mora, C. (2014). Economic and environmental assessment of reusable plastic containers: A food catering supply chain case study. *International Journal of Production Economics, 152,* p 88-101. Retrieved on November 20 2020, from https://doi.org/10.1016/j.ijpe.2013.12.014.
3. Akçalı, E., Çetinkaya, S., Üster, H. (2009) Network design for reverse and closed-loop supply chains: An annotated bibliography of models and solution approaches. Networks, 53 (3) (2009), pp. 231-248. Retrieved on March 15 2021, from https://doi.org/10.1002/net.20267.
4. Akkers, D. H. J. H. (2012). *Optimization of reverse logistics by cooperation between export companies in the floricultural sector.* Retrieved on November 20 2020, from https://research.tue.nl/en/studentTheses/optimization-of-reverse-logistics-by-cooperation-between-export-c.
5. Amin, S. H., Wu, H., Karaphillis, G. (2018). A perspective on the reverse logistics of plastic pallets in Canada. *Journal of Remanufacturing, 2018,* p 153–174. Retrieved on December 4 2020, from https://doi.org/10.1007/s13243-018-0051-0.
6. Andersson, H., Hoff, A., Christiansen, M., Hasle, G., Lokketangen, A. (2010). Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research, 37 (9),* p 1515-1536. Retrieved on March 30 2021, from https://doi.org/10.1016/j.cor.2009.11.009.
7. Archetti, C., Bertazzi, L., Hertz, A., et al. (2012). A hybrid heuristic for an inventory routing problem. *Journal on Computing (2012), 24 (1),* p 101-116. Retrieved on February 14 2021, from https://doi.org/10.1287/ijoc.1100.0439.
8. Bemporad, A., Giua, A., Seatzu, C. (1976). *A Master-Slave Algorithm for the Optimal Control of Continuous-Time Switched Affine Systems.* Retrieved on August 14 2020, from http://cse.lab.imtlucca.it/~bemporad/publications/papers/cdc02-masterslave.pdf.
9. Bertazzi, L., Coelho, L., De Maio, A., Lagana, D. (2019). A matheuristic algorithm for the multi-depot inventory routing problem. *Logistics and Transportation Review (2019), 122,* p 524-544. Retrieved on February 24 2021, from https://doi.org/10.1016/j.tre.2019.01.005
10. Bertazzi, L., Coelho, L., Guerriero, F., Lagana, D. (2013). A stochastic inventory routing problem with stock-out. *Emerging technologies (2013), 27,* p 89-107. Retrieved on February 24 2021, from https://doi.org/10.1016/j.trc.2011.06.003
11. Bertazzi, L., Savelsbergh, M., Speranza, M. G. (2008). Inventory routing. *The Vehicle Routing Problem: Latest Advances and New Challenges,* p. 49-72. Retrieved on April 12 2021, from https://link.springer.com/chapter/10.1007/978-0-387-77778-8_3.
12. Bottani, E., Montanari, R., Rinaldi, M., Vinagli, G. (2015). Modeling and multi-objective optimization of closed loop supply chains: A case study. *Computers & Industrial Engineering 87 (2015)*, p 328–342. Retrieved on January 11 2021, from http://dx.doi.org/10.1016/j.cie.2015.05.009.
13. Bowman, P., Ng, J., Harrison, M., Illic, A. (2009). Building Radio frequency IDentification for the Global Environment: *Reusable Asset Management Model.* Retrieved on March 15 2021, from http://www.bridge-project.eu/data/File/BRIDGE_WP04_RFID_Network_Confidentiality.pdf.

14. Carrano, A. L., Pazour, J. A., Roy, D., Thorn, B. K. (2015). Selection of pallet management strategies based on carbon emissions impact. *Int. J. Production Economics, 164,* p 258-270. Retrieved on December 4 2020, from http://dx.doi.org/10.1016/j.ijpe.2014.09.037.
15. Carrano, A. L., Thorn, B. K., Woltag, H. (2014). Characterizing the Carbon Footprint of Wood Pallet Logistics. *Forest products journal, 64,* p 232-241. Retrieved on December 4 2020, from https://doi.org/10.13073/FPJ-D-14-00011.
16. Cobb, B. R. (2016). Inventory control for returnable transport items in a closed-loop supply chain. *Transportation Research Part E: Logistics and Transportation Review, 86 (2016),* p 53-68. Retrieved on March 26 2021, from https://doi.org/10.1016/j.tre.2015.12.010.
17. Dorner, N., Gassmann, O., Gebauer, H. (2011). Service innovation: Why is it so difficult to accomplish? *Journal of Business Strategy, 32 (3),* p 37-46. Retrieved on April 15 2021, from https://doi.org/10.1108/02756661111121983.
18. Dror, M., Ball, M., Golden. (1985). A computational comparison of algorithms for the inventory routing problem. *Annals of Operations Research, 4 (6),* p 3-23. Retrieved on February 12 2021, from https://link.springer.com/content/pdf/10.1007/BF02022035.pdf.
19. Ech-Charrat, M., Amechnoue, K., Zouadi, T. (2017). Dynamic Planning of Reusable Containers in a Close-loop Supply Chain under Carbon Emission Constraint. *International Journal of Supply and Operations Management, 2017 (4),* p 279-297. Retrieved on March 19 2021, from http://www.ijsom.com/.
20. Elia, V., Gnoni, M. G. (2015). Designing an effective closed loop system for pallet management. International Journal of Production Economics, 170, Part C, p 730-740. Retrieved on December 4 2020, from https://doi.org/10.1016/j.ijpe.2015.05.030.
21. Fazi, S., Fransoo, J. C., Woensel, T. van, Dong, J. (2020). A variant of the split vehicle routing problem with simultaneous deliveries and pickups for inland container shipping in dry-port based systems. *Transportation Research Part E, 142 (2020).* Retrieved on March 19 2021, from https://doi.org/10.1016/j.tre.2020.102057.
22. Figge, F., Hahn, T. (2012). Is green and profitable sustainable? Assessing the trade-off between economic and environmental aspects. *International Journal of Production Economics, 140 (1),* p 92-102. Retrieved on April 15 2021, from https://doi.org/10.1016/j.ijpe.2012.02.001.
23. Floudas, C. A., Lin, X. (2015). Mixed Integer Linear Programming in Process Scheduling: Modeling, Algorithms, and Applications. *Annals of Operations Research, 139,* p 131-162. Retrieved on August 12 2021, from http://cheresearch.engin.umich.edu/lin/publications/Mixed%20Integer%20Linear%20Programming%20in%20Process%20Scheduling.pdf.
24. Glock, C. H. (2017). Decision support models for managing returnable transport items in supply chains: A systematic literature review. *International Journal of Production Economics, 183, Part B,* p 561-569. Retrieved on December 4 2020, from https://doi.org/10.1016/j.ijpe.2016.02.015.
25. Gong, I., Lee, K., Kim, J., Min, Y., Shin, K. (2020). Optimizing Vehicle Routing for Simultaneous Delivery and Pick-Up Considering Reusable Transporting Containers: Case of Convenience Stores. Retrieved on March 13 2021, from https://doi.org/10.3390/app10124162.
26. Govindan, K., Soleimani, H., Kannan, D. (2015). Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research, 240 (3),* p 603-626. Retrieved on March 15 2021, from https://doi.org/10.1016/j.ejor.2014.07.012.

27. Guimaraes, T. A., Coelho, L. C., Schenekember, C. M., Scarpin, C. T. (2018). The two-echelon multi-depot inventory-routing problem. *Computers and Operations Research, 101 (2019),* p 220-233. Retrieved on March 23 2021, from https://doi.org/10.1016/j.cor.2018.07.024.
28. Hahn, T., Figge, F., Pinkse, J., Preuss, L. (2010). Trade-offs in Corporate Sustainability: You Can't Have Your Cake and Eat It, *Business Strategy and the Environment, 19 (2010),* p 217-229. Retrieved on April 15 2021, from https://doi.org/10.1002/bse.674.
29. Hamacher, H., Pedersen, C., Ruzika, S. (2005). *Multiple Objective Minimum Cost Flow Problems: A Review.* Retrieved on June 15 2021, from https://kluedo.ub.uni-kl.de/frontdoor/deliver/index/docId/1677/file/hamacher_ruzika_nr94.pdf
30. Hellström, D. (2009). The cost and process of implementing RFID technology to manage and control returnable transport items, *International Journal of Logistics: Research and Applications, (12) 1,* p 1-21, Retrieved on March 12 2021, from https://doi.org/10.1080/13675560802168526.
31. Hellström, D., Johansson, O. (2010). The impact of control strategies on the management of returnable transport items. *Transportation Research: Part E, 46 (6)*, p 1128-1139. Retrieved on March 15 2021, from https://doi.org/10.1016/j.tre.2010.05.006.
32. Iassinovskaia, G., Rimbourg, S., Riane, F. (2016). *The inventory-routing problem of returnable transport items with time windows and simultaneous pickup and delivery in closed-loop supply chains.* Retrieved on March 15 2021, from https://orbi.uliege.be/bitstream/2268/199421/1/PDIRPTWManuscript.pdf.
33. ISO. (2014). ISO/IEC 15459-5:2014. *Information Technology – Unique Identifiers – Part 5: Unique Identifier for Returnable Transport Items (RTIs). 15459-5.* Retrieved on March 26 2021, from https://www.iso.org/standard/54785.html.
34. Kotnyek, B. (2003). *An annotated overview of dynamic network flows.* Retrieved on June 16 2021, from https://hal.inria.fr/inria-00071643/document.
35. Lefebvre, J. M., Yue, D. (2012). Tracking and Fleet Optimization of Reusable Transport Items in the Shipping Industry. Retrieved on March 15, from https://dspace.mit.edu/bitstream/handle/1721.1/77539/827217640-MIT.pdf?sequence=2&isAllowed=y.
36. Lau, H., Liu, Q., Ono, H. (2002). *Integrating Local Search and Network Flow to Solve the Inventory Routing Problem.* Retrieved on August 14 2020, from https://www.aaai.org/Papers/AAAI/2002/AAAI02-002.pdf.
37. Lau, H., Lim, A., Liu, Q. (2000). *Solving a Supply Chain Optimization Problem Collaboratively.* Retrieved on August 14 2020, from https://www.aaai.org/Papers/AAAI/2000/AAAI00-120.pdf.
38. Liu, G., Li, L., Chen, J., Ma, F. (2020). Inventory sharing strategy and optimization for reusable transport items. *International Journal of Production Economics, 228*. Retrieved on December 4 2020, from https://doi.org/10.1016/j.ijpe.2020.107742.
39. Manfren, M. (2012). Multi-commodity network flow models for dynamic energy management – Mathematical formulation. *Energy Procedia, 14 (2012),* p 1380-1385. Retrieved on June 3 2021, from https://doi.org/10.1016/j.egypro.2011.12.1105.
40. Marler, R., Arora, J. (2009). *The weighted sum method for multi-objective optimization: new insights.* Retrieved on June 13 2020, from https://doi.org/10.1007/s00158-009-0460-7.
41. Mazeika Bilbao, A., Carrano, A. L., Hewitt, M., Thorn, B. K. (2011). On the environmental impacts of pallet management operations. *Management Research Review, 34,* p 1222-1236. Retrieved on December 4 2020, from https://doi.org/10.1108/01409171111178765.

42. Na, B., Sim, M. K., Lee. W. J. (2019). An Optimal Purchase Decision of Reusable Packaging in the Automotive Industry. *Sustainability, 11.* Retrieved on March 25 2021, from https://doi.org/10.3390/sy11236579.
43. Ramkumar, N, Subramanian, P., Narendran, T., Ganesh, K. (2012). Mixed integer linear programming model for multicommodity multi-depot inventory routing problem. OPSEARCH, 49 (4), p 413–429. Retrieved on February 12 2021, from https://doi.org/10.1007/s12597-012-0087-0.
44. Ren, J., Liu, B., Wang, Y. (2017). An optimization model for multi-type pallet allocation over a pallet pool. *Advances in Mechanical Engineering, 9 (5),* p 1–9. Retrieved on March 19 2021, from https://doi.dox.org/10.1177/1687814017705841.
45. Roy, D., Carrano, A. L., Pazour, J. A., Gupta, A. (2016). Cost-effective pallet management strategies. *Transportation Research Part E: Logistics and Transportation Review, 93,* p 358-371. Retrieved on December 4 2020, from https://doi.org/10.1016/j.tre.2016.06.005.
46. Sarkar, B., Ullah, M., Kim, N. (2017). Environmental and economic assessment of closed-loop supply chain with remanufacturing and returnable transport items. *Computers & Industrial Engineering, 111*, p 148-163. Retrieved on December 4 2020, from https://doi.org/10.1016/j.cie.2017.07.003.
47. Serafeim, G. (2018) Public Sentiment and the Price of Corporate Sustainability. Retrieved on August 23 2021, from https://ioandc.com/wp-content/uploads/2018/11/4-ESG_Harvard_Sustainability_Premium_SSRN-id3265502.pdf.
48. Sifaleras, A. (2013). Minimum cost network flows: problems, algorithms, and software. *Yugoslav Journal of Operations Research, 23 (1),* p 3-17. Retrieved on June 16 2021, from https://doi.org/10.2298/YJOR121120001S.
49. Simon, J., Smith, K., West, T. (2010). Price incentives and consumer payment behavior. *Journal of Banking & Finance, 34 (8),* p 1759-1772. Retrieved on August 10 2020, from https://doi.org/10.1016/j.jbankfin.2010.02.001.
50. Tao, Z. G., Guang, Z. Y., Hao, S., Song, H. J., Xin, D. G. (2015). Multi-period closed-loop supply chain network equilibrium with carbon emissions constraints. *Resources, Conservation and Recycling, 104 (2015),* p 354-365. Retrieved on March 15 2021, from http://dx.doi.org/10.1016/j.resconrec.2015.07.016
51. Tornese, F., Pazour, J. A., Thorn, B. K., Roy, D., Carrano, A. L. (2018). Investigating the environmental and economic impact of loading conditions and repositioning strategies for pallet pooling providers. *Journal of Cleaner Production, 172,* p 155-168. Retrieved on December 4 2020, from https://doi.org/10.1016/j.jclepro.2017.10.054.
52. Wei, R., Liu, C. (2020). Research on carbon emission reduction in road freight transportation sector based on regulation-compliant route optimization model and case study. *Sustainable Computing: Informatics and Systems, 28.* Retrieved on December 4 2020, from https://doi.org/10.1016/j.suscom.2020.100408.
53. Xiao, Y., Faff, R., Gharghori, P. (2013). An Empirical Study of the World Price of Sustainability. *Journal of Bus Ethics, 144 (2013),* p 297-310. Retrieved on April 6 2021, from https://doi.org/10.1007/s10551-012-1342-2.
54. Yang, G., Wang, Z., Li, X. (2008). The optimization of the closed-loop supply chain network. *Transportation Research Part E, 45 (2009),* p 16–28. Retrieved on March 15 2021, from https://doi.org/10.1016/j.tre.2008.02.007.

Appendices

Appendix A – Literature on inventory routing problems

From a more operational point of view, Inventory routing problems (IRPs) have been researched in numerous forms, as can be seen in literature overviews by Coelho et al. (2014) and Andersson et al. (2010) which can be found in the Appendix. The objective of inventory routing problems is to determine relocation policies that minimize the total cost, i.e., the sum of inventory holding and transportation costs, while avoiding stock-outs and respecting depot capacity limitations (Bertazzi et al., 2013).

Both Coelho et al. (2014) and Andersson (2010) have reviewed inventory routing problems based on the time horizon, structure, routing, inventory policy and fleet configuration. The structure of the inventory routing problem is highly dependent on the system at hand, as the three options for structure are one-to-one, one-to-many and many-to-many where they resemble distributors-to-customers. Inventory routing problems with the many distributors to many customers structure can also be labeled as multi-depot inventory routing problem. The relocation of stock within an RTI pooler network is therefore identified as a multi-depot inventory routing problem. Within the inventory routing problems, Archetti et al. (2012) have reviewed two different inventory policies, namely Maximum level (ML) and Order-up-to level (OU). ML policy indicates that customers can take any order at any time as long as it stays below the maximum inventory level, while OU assumes customers to be supplied through ordered quantities. Archetti et al. (2012) have concluded that the Order-up-to level inventory policy is more effective at obtaining optimal model solutions than Maximum level inventory policy and will therefore be used in this research.

Guimaraes et al. (2019) have reviewed the multi-depot inventory routing problem based on a two-echelon context. Their approach involved splitting up the problem into smaller subproblems such as the vehicle routing and the input pickups, which resulted in a combination of models and algorithms. Bertazzi et al. (2019) have constructed an algorithm for a multi-depot inventory routing problem with fixed depots but flexible customer allocation per depot. Other inventory routing problems can be found in various sectors outside of the RTI sector, of which an overview of IRPs in literature can be seen in Table 1.

Table 16. Literature overview of inventory routing problems

	Time horizon		Structure		Inventory policy	
Author	Finite	Infinite	One-to-many	Many-to-many	Maximum level	Order-up-to level
Archetti et al. (2012)	✓		✓		✓	✓
Bertazzi et al. (2013)	✓		✓			✓
Bertazzi et al. (2019)	✓			✓	✓	
Coelho et al. (2013)	✓		✓		✓	✓
Guimarães et al. (2019)	✓			✓	✓	✓
Hewitt et al. (2013)	✓			✓	✓	
Michel et al. (2011)	✓		✓			✓
Qamsari et al. (2017)	✓			✓		✓
Raa et al. (2017)		✓	✓		✓	
Ramkumar et al. (2012)	✓			✓		✓
Savelsbergh et al. (2008)	✓			✓	✓	
Zhao et al. (2008)		✓	✓		✓	

With regards to this research, inventory routing problems have rarely been implemented in reverse logistics. Applications to reverse supply chains within closed loop networks are even more scarce, which is the knowledge gat that this thesis aims to fill. Apart from the niche in the sector of application, the IRP itself is different as well. As the inventory levels at depots constantly change, the locations of forecasted stock-outs and allocation of excessive stock within the network is continuously changing. Existing IRP studies focus on a fixed situation of distributors and customers, which cannot be applied in this thesis.

From a modeling point of view, the model that is proposed in this research is a variation of the multi-depot inventory routing with an adaptive set of customers and suppliers. The RTI pooler's activities within the network consist of a multi-depot inventory routing that requires continuous adaptation to the current situation, while looking to predict the upcoming situation.

Appendix A.1 – Literature overview of Coelho et al (2014)

A highly specific literature overview of Inventory Routing Problems has been created by Coelho et al in 2014, which can be used for future references if necessary.

	Time horizon		Structure			Routing			Inventory policy		Inventory decisions			Fleet composition		Fleet size		
Reference	Finite	Infinite	One-to-one	One-to-many	Many-to-many	Direct	Multiple	Continuous	Maximum level (ML)	Order-up-to level (OU)	Lost sales	Backlogging	Nonnegative	Homogeneous	Heterogeneous	Single	Multiple	Unconstrained
Bell et al. (1983)	✓			✓			✓		✓		✓				✓		✓	
Burns et al. (1985)	✓			✓		✓	✓		✓		✓			✓				✓
Dror, Ball, and Golden (1985)	✓			✓			✓			✓	✓			✓			✓	
Dror and Levy (1986)	✓			✓			✓		✓				✓	✓			✓	
Dror and Ball (1987)	✓		✓	✓			✓		✓		✓			✓			✓	
Chien, Balakrishnan, and Wong (1989)	✓			✓			✓		✓			✓			✓		✓	
Anily and Federgruen (1990)		✓		✓			✓		✓		✓			✓				✓
Gallego and Simchi-Levi (1990)		✓		✓		✓			✓		✓			✓				✓
Campbell et al. (1998)	✓			✓			✓		✓					✓				✓
Christiansen (1999)	✓				✓			✓	✓		✓				✓		✓	
Bertazzi, Paletta, and Speranza (2002)	✓			✓			✓			✓			✓	✓		✓		
Ronen (2002)	✓				✓			✓	✓		✓			✓				✓
Ribeiro and Lourenço (2003)	✓			✓			✓		✓		✓			✓				✓
Abdelmaguid (2004)	✓			✓			✓		✓			✓		✓			✓	
Campbell and Savelsbergh (2004)	✓			✓			✓		✓		✓			✓			✓	
Abdelmaguid and Dessouky (2006)	✓			✓			✓		✓			✓		✓			✓	
Aghezzaf, Raa, and van Landeghem (2006)		✓		✓			✓		✓		✓			✓			✓	

Figure 19. IRP literature overview part one

	Time horizon		Structure			Routing			Inventory policy		Inventory decisions			Fleet composition		Fleet size		
Reference	Finite	Infinite	One-to-one	One-to-many	Many-to-many	Direct	Multiple	Continuous	Maximum level (ML)	Order-up-to level (OU)	Lost sales	Backlogging	Nonnegative	Homogeneous	Heterogeneous	Single	Multiple	Unconstrained
Archetti et al. (2007)	✓			✓			✓		✓	✓			✓	✓		✓		
Raa and Aghezzaf (2008)		✓		✓			✓		✓				✓	✓			✓	
Savelsbergh and Song (2008)	✓				✓			✓	✓		✓			✓			✓	
Zhao, Chen, and Zang (2008)		✓		✓			✓		✓				✓	✓				✓
Abdelmaguid, Dessouky, and Ordóñez (2009)	✓			✓			✓		✓			✓			✓		✓	
Boudia and Prins (2009)	✓			✓			✓		✓				✓	✓			✓	
Raa and Aghezzaf (2009)		✓		✓			✓		✓		✓			✓			✓	
Geiger and Sevaux (2011a)	✓			✓			✓		✓		✓			✓				✓
Solyalı and Süral (2011)	✓			✓			✓			✓			✓	✓		✓		
Adulyasak, Cordeau, and Jans (2013)	✓			✓			✓		✓	✓			✓	✓	✓		✓	
Archetti et al. (2012)	✓			✓			✓		✓	✓			✓	✓		✓		
Coelho, Cordeau, and Laporte (2012a)	✓			✓			✓		✓	✓			✓	✓		✓		
Coelho, Cordeau, and Laporte (2012b)	✓			✓			✓		✓	✓			✓	✓			✓	
Michel and Vanderbeck (2012)	✓			✓			✓			✓			✓	✓			✓	
Coelho and Laporte (2013a)	✓			✓			✓		✓	✓			✓	✓	✓		✓	
Coelho and Laporte (2013b)	✓			✓			✓		✓	✓			✓	✓			✓	
Hewitt et al. (2013)	✓				✓			✓	✓				✓		✓		✓	

Figure 20. IRP literature overview part two

Appendix A.2 – Literature overview of Andersson et al. (2010)

		Year	Demand/topology/routing	Inventory/fleet composition/fleet size	Approach
1	Blumenfeld et al. [29]	1985	Deterministic/one-to-many, many-to-many/direct	Fixed/homogeneous/unconstrained	Analytical method
2	Anily and Federgruen [9]	1990	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Lower bound, fixed partitions
3	Gallego and Simchi-Levi [65]	1990	Deterministic/one-to-many/direct	Fixed/homogeneous/unconstrained	Lower bound, analytical
4	Anily and Federgruen [12]	1993	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Lower bound, fixed partitions
5	Minkoff [92]	1993	Stochastic/one-to-many/multiple	Lost sale/homogeneous/unconstrained	Markov decision process, customer decomposition
6	Speranza and Ukovich [113]	1994	Deterministic/one-to-one/direct	Fixed/homogeneous/unconstrained	Lower and upper bounds, mixed integer programs
7	Bramel and Simchi-Levi [30]	1995	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Lower bound, fixed partition, cluster first/route second
8	Barnes-Schuster and Bassok [19]	1997	Stochastic/one-to-many/direct	Back-order/homogeneous/unconstrained	Lower bound, analytical
9	Bertazzi et al. [26]	1997	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Sequential heuristic, single customer analysis, local search
10	Herer and Roundy [80]	1997	Deterministic/one-to-many/multiple	Fixed/homogeneous/single	Submodular approximation, local search
11	Jones and Qian [85]	1997	Deterministic/one-to-many/direct	Fixed/homogeneous/unconstrained	Lower bound, analytical
12	Viswanathan and Mathur [123]	1997	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Constructive insertion heuristic
13	Chan and Simchi-Levi [38]	1998	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Lower bound, fixed partitions
14	Qu et al. [97]	1999	Stochastic/one-to-many/multiple	Stock-out/homogeneous/single, uncapacitated	Lower bound, modified periodic policy, routing/inventory decomposition
15	Reiman et al. [101]	1999	Stochastic/one-to-many/direct, multiple	Back-order/homogeneous/single	Heavy traffic analysis, Monte-Carlo simulation
16	Kleywegt et al. [88]	2002	Stochastic/one-to-many/direct	Lost sale/homogeneous/multiple	Markov decision process, iterative heuristic
17	Adelman [2]	2003	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Markov decision process, linear reformulation, dual information
18	Adelman [3]	2004	Stochastic/one-to-many/multiple	Lost sale/homogeneous/multiple	Markov decision process, linear reformulation, dual information
19	Kleywegt et al. [89]	2004	Stochastic/one-to-many/multiple	Lost sale/homogeneous/multiple	Markov decision process, iterative heuristic
20	Aghezzaf et al. [4]	2006	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Heuristic column generation
21	Custódio and Oliveira [51]	2006	Stochastic/one-to-many/multiple	Fixed/heterogeneous/multiple	Constructive insertion heuristic
22	Schwartz et al. [109]	2006	Stochastic/one-to-many/multiple	Back-order/homogeneous/single	Analytical model, simulation, change-revert heuristic
23	Jung and Mathur [86]	2007	Deterministic/one-to-many/multiple	Fixed/homogeneous/unconstrained	Lower bound, fixed partition integer linear program
24	Stacey et al. [115]	2007	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Cluster first/route second location based heuristic
25	Hvattum and Løkketangen [82]	2008	Stochastic/one-to-many/multiple	Stock-out/homogeneous/multiple	Markov decision process, scenario tree, GRASP
26	Hvattum et al. [83]	2008	Stochastic/one-to-many/multiple	Stock-out/homogeneous/multiple	Markov decision process, scenario tree, progressive hedging
27	Li et al. [90]	2008	Deterministic/one-to-many/direct	Fixed/homogeneous/single	Construction algorithm
28	Raa and Aghezzaf [98]	2008	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Heuristic column generation
29	Raa and Aghezzaf [99]	2009	Deterministic/one-to-many/multiple	Fixed/homogeneous/multiple	Heuristic column generation

Figure 21. Literature overview IRP

Appendix B – Relocation flows over time

Since the outcome of the inventory management optimization model is defined by flows of certain commodities between depots over time, the results are difficult to interpret when looking at the flows themselves. For the scheduling of relocations within RTI pooler networks however, these are the results that are most interesting for the planning phase. The results in data frame format can be seen below in Figure 22.

In [19]:
```
# Choose the period t in 0 - 52
# n_df_list[t]

n_df_list[0]
```

Out[19]:

	0 -> 1	0 -> 2	0 -> 3	0 -> 4	0 -> 5	0 -> 6	0 -> 7	0 -> 8	0 -> 9	0 -> 10	...	59 -> 49	59 -> 50	59 -> 51	59 -> 52	59 -> 53	59 -> 54	59 -> 55	59 -> 56	59 -> 57	59 -> 58
0	0	0	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

2 rows × 3540 columns

In [20]:
```
# Choose the period t in 0 - 52
# flow_df_list[t]

flow_df_list[0]
```

Out[20]:

	0 -> 1	0 -> 2	0 -> 3	0 -> 4	0 -> 5	0 -> 6	0 -> 7	0 -> 8	0 -> 9	0 -> 10	...	59 -> 49	59 -> 50	59 -> 51	59 -> 52	59 -> 53	59 -> 54	59 -> 55	59 -> 56	59 -> 57	59 -> 58
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	0	0	2000	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

6 rows × 3540 columns

Figure 22. Relocation flows between depots

The full set of results regarding any of the models, including possible added policies, are available on request at: jasperendlich@gmail.com or j.endlich@student.tudelft.nl.

Appendix C – Code

Master – Network flow model

```
// General initialisation of the model
{int} Nodes = ...;
range Depots = 1..56;
range RepairLocations = 57..60;
{int} Periods = ...;
{int} Periods0 = ...;
{int} Commodities = ...;
{int} WorkingInventory = ...;
float Alpha = ...;
float Beta = ...;
float Gamma = ...;
int StockOutPrice = ...;
int M = ...;

// These values are given parameters to the model, defined in the .dat file
int InitialSupply[Nodes][WorkingInventory][Commodities] = ...;
int DepotCapacity[Nodes][Commodities] = ...;
int Demand[Nodes][WorkingInventory][Commodities][Periods] = ...;
int Returns[Nodes][WorkingInventory][Commodities][Periods] = ...;
int MaxRepairRate[Nodes][Commodities] = ...;
float HandlingCost[Nodes][Commodities] = ...;

// Create a record to hold information about each arc
tuple arc {
   key int fromnode;
   key int tonode;
   float distance;
   float cost;
   int samedepot;
}

// Get the set of arcs
{arc} Arcs = ...;

// Here's the decision variable
dvar int+ Inventory[Nodes][WorkingInventory][Commodities][Periods0];
dvar int+ Flow[a in Arcs][WorkingInventory][Commodities][Periods0];
dvar int+ StockOutCost[Nodes][Periods];
dvar int+ RepairRate[Nodes][Commodities][Periods];

// The objective is to minimize the total distance and/or total costs, depending on the trade-off
dexpr float TotalDistance = sum (a in Arcs, f in WorkingInventory, k in Commodities, t in Periods)
a.distance * Flow[a][f][k][t];
dexpr float TotalTransportCost = sum (a in Arcs, f in WorkingInventory, k in Commodities, t in Periods)
a.cost * Flow[a][f][k][t];
dexpr float TotalHandlingCost = sum(a in Arcs, f in WorkingInventory, k in Commodities, t in Periods)
        Flow[a][f][k][t] * (HandlingCost[a.fromnode][k] + HandlingCost[a.tonode][k]) * a.samedepot;
dexpr float TotalCost = TotalTransportCost + TotalHandlingCost;
dexpr int TotalStockOutCost = sum (i in Nodes, t in Periods) StockOutCost[i][t] * StockOutPrice;

dexpr float Total = Alpha * TotalDistance + Beta * TotalCost + Gamma * TotalStockOutCost;
minimize
  Total;

// Constraints section
subject to {
  // First we start with the initial inventory
  forall(k in Commodities, f in WorkingInventory, i in Nodes)
    ctInit:
        Inventory[i][f][k][0] == InitialSupply[i][f][k];

  // Inventory calculations are performed below
```

```
  // The inventory definition at normal depots is calculated based on the previous inventory, the inflow
and the outflow
  forall(k in Commodities, i in Depots, t in Periods)
        ctWorkingInventory:
        Inventory[i][1][k][t] == Inventory[i][1][k][t-1] + Returns[i][1][k][t] + Demand[i][1][k][t] +
RepairRate[i][k][t]
        + sum(j in Nodes, <j,i,d,c,s> in Arcs) Flow[<j,i,d,c,s>][1][k][t-1]
        - sum(j in Nodes, <i,j,d,c,s> in Arcs) Flow[<i,j,d,c,s>][1][k][t];

  // The inventory definition at normal depots is calculated based on the previous inventory, the inflow
and the outflow
  forall(k in Commodities, i in Depots, t in Periods)
        ctBrokenInventory:
        Inventory[i][0][k][t] == Inventory[i][0][k][t-1] + Returns[i][0][k][t] + Demand[i][0][k][t] -
RepairRate[i][k][t]
        + sum(j in Nodes, <j,i,d,c,s> in Arcs) Flow[<j,i,d,c,s>][0][k][t-1]
        - sum(j in Nodes, <i,j,d,c,s> in Arcs) Flow[<i,j,d,c,s>][0][k][t];

  // The inventory definition at repair depots is calculated based on the previous inventory, the newly
repaired inventory and the outflow
  forall(k in Commodities, i in RepairLocations, t in Periods)
        ctWorkingRepair:
        Inventory[i][1][k][t] == Inventory[i][1][k][t-1] + RepairRate[i][k][t]
        + sum(j in Nodes, <j,i,d,c,s> in Arcs) Flow[<j,i,d,c,s>][1][k][t-1]
        - sum(j in Nodes, <i,j,d,c,s> in Arcs) Flow[<i,j,d,c,s>][1][k][t];

  // The inventory definition at repair depots is calculated based on the previous inventory, the inflow
minus the repaired inventory
  forall(k in Commodities, i in RepairLocations, t in Periods)
        ctBrokenRepair:
        Inventory[i][0][k][t] == Inventory[i][0][k][t-1] - RepairRate[i][k][t]
        + sum(j in Nodes, <j,i,d,c,s> in Arcs) Flow[<j,i,d,c,s>][0][k][t-1];

  // Initializing the safety stock is important for the prevention of stock-out
  forall(f in WorkingInventory, k in Commodities, i in Nodes, t in Periods)
    ctMinimum:
        Inventory[i][f][k][t] >= 0;

  // Let's try to make a stock-out variable
  forall(k in Commodities, i in Depots, t in Periods)
    ctStockOut:
        Inventory[i][1][k][t] >= 200 - M * StockOutCost[i][t];

        forall(i in Nodes, t in Periods)
          ctZ:
                0 <= StockOutCost[i][t] <= 1;

  // Initializing the maximum level of stock is important for the feasibility of the model
  forall(k in Commodities, i in Nodes, t in Periods)
    ctMaximum:
        sum(f in WorkingInventory) Inventory[i][f][k][t] <= DepotCapacity[i][k];

  // Repair rate is not fixed, but can fluctuate because of too low stock. The goal is to repair as much
as possible
  forall(i in Nodes, k in Commodities, t in Periods)
    ctRepair:
        MaxRepairRate[i][k] * 0.95 <= RepairRate[i][k][t] <= MaxRepairRate[i][k];
```

Slave – Multi-knapsack model

```
// General initialisation of the model
{int} Nodes = ...;
{int} Periods = ...;
{int} Periods0 = ...;
{int} Commodities = ...;
{int} WorkingInventory = ...;
float Alpha = ...;
float Beta = ...;
float Gamma = ...;
int StockOutCost = ...;
int Ctruck = ...;

// Create a record to hold information about each arc
tuple arc {
   key int fromnode;
   key int tonode;
   float distance;
   float cost;
   int samedepot;
}

// Get the set of arcs
{arc} Arcs = ...;

// These values are given parameters to the model, defined in the .dat file
float HandlingCost[Nodes][Commodities] = ...;
int Flow[Arcs][WorkingInventory][Commodities][Periods0] = ...;
float Weights[Commodities] = ...;

// Here's the decision variable
dvar int+ n[a in Arcs][WorkingInventory][Periods];

// The objective is to minimize the total distance and/or total costs, depending on the trade-off
dexpr float TotalDistance = sum (a in Arcs, f in WorkingInventory, t in Periods) a.distance * n[a][f][t];
dexpr float TotalTransportCost = sum (a in Arcs, f in WorkingInventory, t in Periods) a.cost * n[a][f][t];
dexpr float TotalHandlingCost = sum(a in Arcs, f in WorkingInventory, k in Commodities, t in Periods)
        Flow[a][f][k][t] * (HandlingCost[a.fromnode][k] + HandlingCost[a.tonode][k]) * a.samedepot;
dexpr float TotalCost = TotalTransportCost + TotalHandlingCost;
dexpr int TotalStockOutCost = StockOutCost;

dexpr float Total = Alpha * TotalDistance + Beta * TotalCost + Gamma * TotalStockOutCost;
minimize
  Total;

// Constraints section
subject to {
  // Normal stacking
  // Not overloading the capacity of all trucks
  forall(f in WorkingInventory, a in Arcs, t in Periods)
        ctTruckLoad:
                sum(k in Commodities) Flow[a][f][k][t] * Weights[k] <= n[a][f][t] * Ctruck;
}
```

Slave – Depot capacity optimization

```
// General initialisation of the model
range Depots = 1..60;
range Periods = 0..53;
{int} Commodities = ...;
{int} WorkingInventory = ...;
float SafetyRatio = ...;

// These values are given parameters to the model, defined in the .dat file
int Inventory[Depots][WorkingInventory][Commodities][Periods] = ...;
int CurrentCapacity[Depots] = ...;
float DepotCost[Depots] = ...;

// Here's the decision variable, for the amount of square footage available
dvar float+ Capacity[Depots];
dvar float+ MaxInventory[Depots][Commodities];

// The objective is to minimize the total distance and/or total costs, depending on the trade-off
dexpr float OldCost = sum(i in Depots) DepotCost[i] * CurrentCapacity[i];
dexpr float TotalCost = sum (i in Depots) DepotCost[i] * Capacity[i];

minimize TotalCost;

subject to {
  forall(i in Depots, k in Commodities)
        ctInvToSize:
                MaxInventory[i][k] == max(t in Periods) sum(f in WorkingInventory) Inventory[i][f][k][t];

  forall(i in Depots)
    ctConfigAll:
        Capacity[i] == ((MaxInventory[i][3] / 30 * 0.76) +
        ((MaxInventory[i][2] - (4 * MaxInventory[i][3] / 30)) / 200 * 1.05) +
        ((MaxInventory[i][1] - ((MaxInventory[i][2] - (4 * MaxInventory[i][3] / 30)) / 200) -
(MaxInventory[i][3] / 30))) / 15 * 0.76) * SafetyRatio;
}
```