RENSA
FAMILY COMPANY

Master Thesis

The integration of crossdocking facilities in an existing outbound logistics network

08-04-2024

Auke Rasing

University of Twente
MSc Industrial Engineering and Management
Specialization Production and Logistics Management

Company
Rensa Family Company
Beekseweg 3
6942 JC Didam

Educational institution
University of Twente
Drienerlolaan 5
7522 NB Enschede

Examination committee

First supervisor:	dr.ir. W.J.A. Van Heeswijk (Wouter)
Second supervisor:	dr.ir. L.L.M. Van der Wegen (Leo)
Company supervisor:	Bob Lammers

Preface

This thesis marks the conclusion of my time as a master's student in Industrial Engineering and Management at the University of Twente.

First of all, I want to thank Rensa Family Company for providing me the opportunity to carry out this research and the support I received from people throughout the organization. Special thanks go to my internal supervisor Bob Lammers for his enthusiasm and support, which made me feel welcome immediately, and which is one of the reasons I am working at Rensa today.

I would also like to thank my University of Twente supervisor, Wouter van Heeswijk for lots of valuable feedback, helping me to shape and give direction to this research. Lastly, I want to thank Leo van der Wegen for his expertise and criticism that helped finalizing this thesis report.

I wish you a pleasant reading experience.

Auke Rasing

April 8, 2024, Arnhem

Management summary

Rensa Family Company is a facilitating company within the Rensa group, operating in technical wholesaling in the installation market. This research was triggered by growth of Rensa and the pressure that this growth puts on its logistical processes.

Rensa's current logistics network consists of five warehouses. All warehouse have a storage function, but only one of them serves as outbound logistics warehouse. Order picking takes places at all warehouses and goods are transported to the outbound warehouse throughout the day. The outbound warehouse is reaching capacity limits, and the other warehouses are not suitable for outbound activities.

In Rensa's logistics, the main focus of this research, two main problems can be observed:

- ***The outbound logistics warehouse lacks capacity in terms of docks doors and floor space***
- ***The logistics network is not suitable for fleet electrification and city center distribution***

Earlier internal research by Groters (2020) concludes that improvement potential within the current logistics infrastructure is limited and insufficient to facilitate the growing organization. The considered alternative that Groters and Rensa see most potential in, consists of integrating crossdocking hubs in the logistics network. The hub forms a location where goods are directly transferred to outgoing vehicles without storing them in between. This leads us to the following research goal:

- *Assess the viability, with a specific focus on transport costs, of locating one or more decentral crossdocking hubs in the logistics network of Rensa.*

Context analysis

A combination of literature review, interviews and a survey, led to factors regarding the implementation of crossdocking hubs that are relevant for Rensa. The most important factors are:

- More flexible and efficient use of truck and driver capacity in a hub network
- Effect of hub capacity on the effectivity and efficiency of an organization on site
- Effect of opening hubs on transport planning

We see that the main effect that Rensa expects from moving to a new network design is the ability to organize transport from hubs more efficiently, having shorter driving distances and the ability to perform a second delivery trip on a day, something that is not done currently. The main concern lies in the fact that we are creating several smaller, separated goods streams by opening crossdocking hubs. Challenges may be in having a small but stable organization on a hub, the ability to do efficient transport planning and dealing with increased demand fluctuations.

Modelling approach

To see if a hub network can be viable for Rensa, we want to compare the current with the new logistics network. We combine location and routing models to model hub locations, goods shuttling between central DCs and hubs and delivery tours from hubs to end customers. To get a view of the number of hubs and their location that looks promising, we first solve a location problem, the strategic issue of locating one or more facilities in a given area. To be able to assess transport costs in the logistics network, we incorporate the operational problem of vehicle routing. The location problem gives promising combinations of hub locations, that form the input of a vehicle routing model.

We solve a p-median location model in which the weighed average distance to customers from a set of candidate locations is minimized. Weighing solution quality and calculation effort, a set of 12 candidate locations was chosen together with the transport department of Rensa. Rensa wants to investigate the possibility of performing second delivery tours from a hub location. We do this by using a savings algorithm followed by a bin packing procedure that tries to combine delivery tours.

Conclusions

Modelling shows that a logistics network using crossdocking hubs is viable for Rensa looking at transport costs. A decrease in operational transport costs of 7% can be achieved when opening two hubs. Results show that two hubs is the minimum number to divide the Netherlands in equal, logical zones. Having more than two hubs may be viable looking at factors like the average distance to customers or city centers, in relation to electrical distribution, or size that individual hubs have. Looking at transport costs only, there is no incentive to open more than two hub locations.

By opening crossdocking hubs, the current outbound logistics warehouse of Rensa can be relieved significantly. The fraction of goods directly distributed from here ranges from 57% to only 23% for one to five hubs. Decreasing transport costs, less planned delivery tours and their increased efficiency show us that we are able to operate trucks from a hub more flexibly and efficiently.

We advise Rensa to take next steps towards implementation of crossdocking hubs in the logistics network, and to do this in an interdepartmental team. Further, we propose a phased implementation with the opening of one hub. This gives possibility to experience the broad operational impacts beyond the scope of transport in this research, while maintaining risk and investment relatively low.

Contents

1. Introduction

This chapter introduces Rensa Family Company and its logistics network. It gives the research motivation and describes the problem identification that leads to a research objective and research questions. Furthermore, it defines the scope of the research and provides a reading guide for the remainder of the report.

1.1 Rensa Family Company

Rensa Family Company is a facilitating company within the Rensa group, a family of 14 companies operating in technical wholesaling in the installation market. Apart from two members of the group that are located in Germany and Romania, the rest of the companies are based in the Netherlands. The group originates from Rensa Heating and Ventilation, but now has related businesses ranging from Gévier bathroom and sanitary to Libra solar. All companies within the Rensa Family can be found in Figure 1. Rensa Family Company performs supporting services for other members in the group, such as logistics services, warehousing, human resources, finance, marketing and sales, technical support, eCommerce and IT. Rensa Family Company was created to centralise supporting tasks, so that knowledge is bundled, scale advantages can be achieved, and operations can be performed more efficiently. The Rensa group has ambitious growth plans, but under this growth, customer service, customer intimacy and reliability remain primary focus.

Figure 1 Companies within the Rensa Family Group (Rensa Family Company)

Several companies within the Rensa group have their supportive functions fully facilitated by Rensa Family Company. These are Rensa Heating and Ventilation, Gévier and Gafco. Others have only some functions carried out by Rensa Family Company or are completely independent organizations. By far the biggest company served by Rensa Family Company is Rensa Heating & Ventilation. It accounts for around 85% of the turnover, when we look at the three companies fully serviced by Rensa Family Company in their supportive functions.

As mentioned in the previous paragraphs, Rensa Family Company performs logistics activities and supportive functions for companies within the Rensa group. This research focuses on the logistical services that Rensa Family Company performs for its sister companies. The logistics departments within Rensa Family Company are Transport, Logistics management and Warehousing, which is divided into Large cargo and Small cargo. Warehousing is responsible for the small cargo warehouse and the five large cargo warehouses of the company, making sure that goods are received and stored, orders are picked and goods are loaded for distribution. The transport department is responsible for getting the products to the customer. The department consists of transport planning, fleet management, service department and truck drivers. The company's fleet consists of more than 80 trucks, and is complemented by a flexible layer of external partners every day. Most of the days, more than a hundred vehicles are on the road to deliver goods for Rensa Family Company. In 2019, the company started with the acquisition of a fleet of transport vans, aimed at delivering smaller packages efficiently on own-account. Rensa handles a broad variety of products, ranging from tubes and pipes and small installation parts to boilers and heat pumps. Heat pumps form a product category that has been rapidly growing over the last years.

In the remainder of this research, when we talk about Rensa Family Company we will just say 'Rensa'.

1.2 Research motivation

This section describes the internal considerations of Rensa that set this research project in motion.

The motivation for this research all starts with growth. Rensa has been growing steadily over the past years and expects this growth to continue in the foreseeable future. Over the past years, this growth was made possible by constantly improving business processes and investing in the organization. Examples of this are putting in use around ten extra docks for night delivery at the outbound logistics warehouse and the opening of a completely new warehouse DC6XL with 16.000m2 of storage space at the start of 2020 (Rensa, 2019).

Although Rensa aims to sustain their growth figures, the company actively propagates the motto of first being the best before becoming the biggest. The logistics departments of Rensa feel that being the best will be at stake over the coming years, as growth puts logistics processes under pressure. Rensa therefore thinks of ways to adapt the organization to keep up with expected future growth.

In this light, internal research on Rensa's logistics network was performed by Groters (2020). He finds that improvement potential within the current logistics infrastructure is limited and that it only forms a short term solution for the growing organization. He advocates to look for opportunities in a different logistics network design of Rensa.

On the first sight, the logical and known option to Rensa would be to build a new outbound warehouse close to its existing warehouses in Doetinchem and Didam, or expand one of these. This would not change the logistics network, still being centralized in the same area. However, Rensa doubts if the current logistics network design is also the best option towards the future. Also, distribution from a single location in the east of the Netherlands does not seem to be the most logical strategy from a geographical point of view.

Rensa wants to research the option of expanding their logistics network in a more decentral manner. The idea that lives within the organization is locating a number of warehouses performing crossdocking throughout the Netherlands. This research opportunity is also mentioned in the research of Groters (2020). These new crossdocking warehouses will be located further away from the current central warehouses in Doetinchem and Didam, making the network more decentralized in that sense. Large truckloads are transported to these new warehouses, and can be consolidated for customer delivery over there. From the new warehouses customer delivery is performed.

Rensa wants to gain better understanding of the impact that locating such decentral crossdocking hubs would have on the organization. Also, they want to know if such a network would be cost effective compared to the current logistics network of Rensa. This current network is outlined in the next section.

1.3 Rensa's logistics network

This section describes how the current logistics network of Rensa and its processes are organized. An overview of the transport and warehousing processes and how they are related is given in Figure 2.

The current outbound logistics network of Rensa, consists of five warehouses:

1. **DC1** Located in Didam, highly automated for storage of small cargo
2. **DC4** Located in Doetinchem, storage of larger cargo and Rensa's outbound logistics warehouse
3. **DC5** Located in Doetinchem, storage of larger cargo, mainly radiators
4. **DC6** Located in Doetinchem, storage of larger cargo, mainly sanitary
5. **DC6XL** Located in Doetinchem, storage of larger cargo, mainly heat pumps

DC2 and DC3 did also exist, but were closed at some point.

The highly automated warehouse DC1 in Didam handles small cargo, the other four warehouses located in Doetinchem handle different kinds of larger products. Inventory is held at all five warehouses. DC4 serves as the outbound logistics warehouse and combines its storage role with a crossdocking function. The basic idea behind crossdocking is to transfer incoming shipments directly to outgoing vehicles without storing them in between (Van Belle, Valckenaers, & Cattrysse, 2012). The guideline used in general, is that all shipments leave the terminal within 24 hours (Konrad & Boysen (2011),Van Bell, Valckenaers & Cattrysse (2012), Buakum & Wisittipanich (2019)).

Orders are sent to Rensa's warehouse management system (WMS) between 7.00AM and 18.00PM and are delivered the next day by default. Goods that are picked at the other DCs than DC4, are loaded in trucks shuttling to DC4. At DC4 the goods are consolidated and goods picked from stock at DC4 are added. Goods are consolidated in areas in front of the docks, based on destination zip code of the

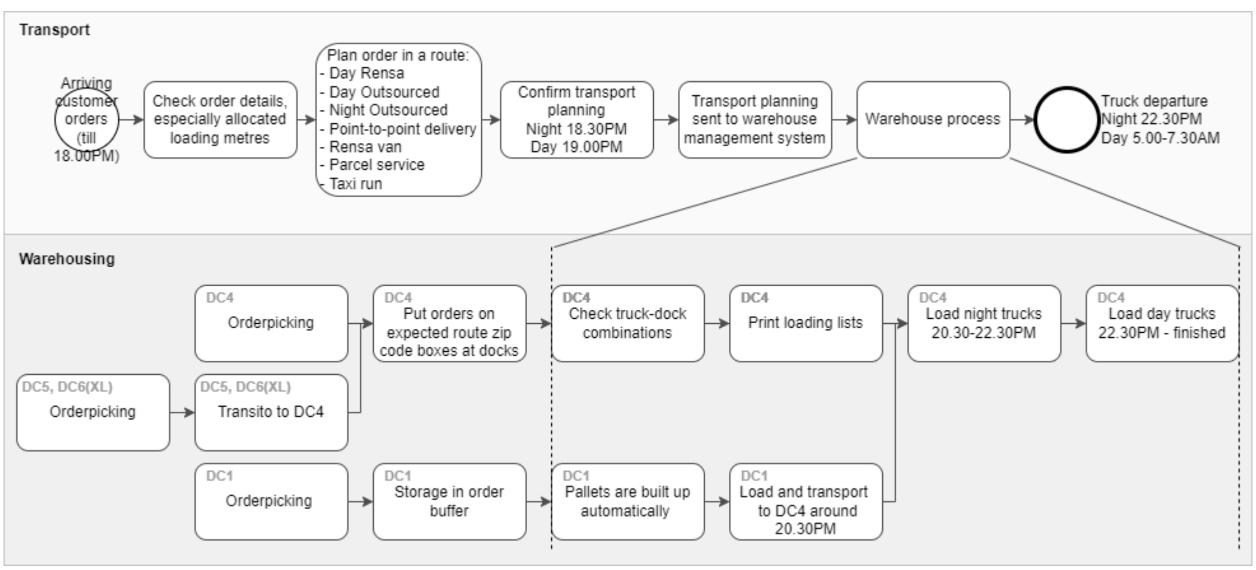


Figure 2 Logistics process overview

customer order they are in. At the end of the day, when the transport planning is complete, goods are loaded in the outbound trucks.

The transport planning department works on route planning throughout the day, and confirms the complete planning at around 19.00PM. At that moment all routes are sent from the transport planning application to the WMS. Goods of Rensa are delivered via two different streams, namely night and day delivery. The night trucks are loaded first and depart from Doetinchem at around 22.30 PM. The day trucks are loaded when the night trucks have departed. They start their delivery routes roughly between 5.00 and 8.00 AM. The vast majority of distribution is performed by Rensa's truck fleet, complemented by flexible capacity from third parties. A fleet of delivery vans is used to distribute small orders. A detailed process flow of Rensa's transport planning process can be found in Appendix B: Transport planning.

Customers of Rensa are distributed all over the Netherlands. To be able to better serve customers that are further away from the DCs in Doetinchem and Didam, Rensa works with seven truck pitches, located in the north, west and south of the Netherlands. A map with all truck pitches can be found in Appendix A: Truck pitches. Part of the trucks in the day delivery, drive to these seven truck pitches. The truck pitch itself is nothing more than a parking lot with a fence. A pool of three local drivers is assigned to one truck pitch. One driver drives with an empty combination of two load compartments from truck pitch to DC4 in Doetinchem. This is done at night. The two compartments are loaded at DC4 and the driver drives back to the truck pitch. In the morning, two other drivers come to the pitch and both drive a delivery route with one load compartment.

One of the advantages for Rensa of using these truck pitches is that local drivers deliver at the customers. These drivers come from the same area, speak the same dialect and know the customers on a personal basis as they always deliver in the same region. Another advantage is that the two drivers that start their delivery route from the truck pitch still have a full working day left. When starting from Doetinchem, they would easily lose half of their time driving from Doetinchem to their delivery area and back. Planners decide which orders will be delivered from the truck pitches based on the specific demand on a day.

The current outbound logistics network is summarized in Figure 3. We see the goods stream from separate DCs to DC4, where consolidation is done and from where transport to customers takes place. Transport either goes in a direct delivery route to customer or via one of the truck pitches, as described in the previous paragraphs.

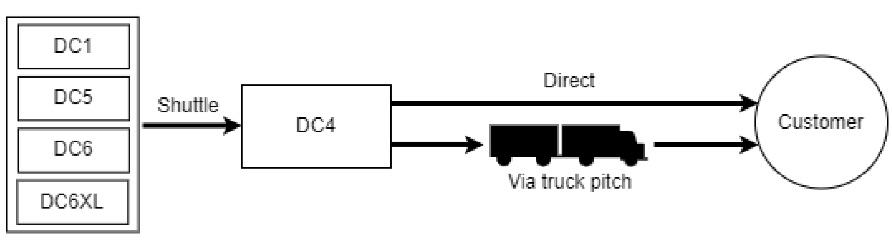


Figure 3 Current outbound logistics network design

1.4 Problem identification

This section dives deeper in the problems and developments that are the drivers for this research. We start with the growth of Rensa as it plays an important role in challenges and problems, as we already saw in the research motivation of Section 1.2. The remainder of the section focuses on problems concerning physical storage space, capacity of DC4 and the sustainable transition.

1.4.1 Growth of Rensa

This research project is driven by the growth that Rensa has achieved over the past years and the growth that is expected for the coming years. Over the year 2020 Rensa experienced a growth of 14% in turnover. For the years 2021-2024 a growth in turnover of around 10% per year on average is expected. Although this growth is a very positive development, it puts the operations within the company under pressure.

In his research, Groters (2020) already points out that the current outbound logistics network of Rensa is reaching its limits. He states that continuing the growth strategy under the current infrastructure, will harm the outstanding performance of the company. This expresses itself in several places as described in subsections 1.4.2, 1.4.3 and 1.4.4.

1.4.2 Physical storage space

With growing sales volumes, the amount of goods that Rensa stores in its warehouses also rises. The physical storage space in the five warehouses of Rensa is limited. Continuously, projects are carried out at Rensa to deal with growing stock volumes.

At small cargo warehouse DC1 the last project, adding 17,000 storage locations with a new automated stocking and picking area, has finished in 2021. The follow up project immediately started, replacing an older automated storage and picking system with the newest standard. This last project expands storage capacity at DC1 from 80,000 to 100,000 storage locations, while also increasing picking capacity and efficiency.

Expansion also took place at DC6XL, the warehouse that became operational in 2020. The warehouse was built with wide aisles, with the idea to be able to transform to a narrow aisle warehouse within about five years. In the end, this transition started in 2021 and was finished mid-2022, only two years after opening the warehouse. Storage capacity of the warehouse was almost doubled, and capacity utilization of the warehouse is now just over 50%.

While physical storage space will remain a challenge for Rensa for the coming years, the organization already has the focus that is needed on this topic. The projects that are or have been undertaken remain a challenge, but result in physical storage space not being a problem that we focus on in this research project.

1.4.3 Capacity of DC4 as outbound logistics warehouse

What will be of main focus in this research, are capacity problems at DC4 as outbound logistics warehouse. In his research, Groters (2020) already predicts that DC4 will most likely be the bottleneck in Rensa's logistics network.

Competition for space

DC4 serves as outbound logistics warehouse. Products from the other DCs are consolidated and loaded in trucks here. Besides its crossdocking function, goods are also stored and picked at DC4. This combined function means that there is always a tradeoff between storage and picking and crossdocking activities. The situation is further complicated by the fact that also the retour and prefabrication departments are situated in DC4. In the growing organization, these departments also see their goods stream increase, and have a continuous demand for extra space.

Reducing storage and picking at DC4 seems like a viable solution for capacity problems. However, this is not a preferable solution. DC4 is the most efficient warehouse to pick from, as consolidation areas at the docks are close to the picking area, and no shuttling from other DCs is needed. However, increased goods flow makes the warehouse more crowded and asks for increased handling space and capacity. There is more pressure on the order picking process, as more orders must be picked, for which more people are needed. It is a challenge to have all orders picked and staged on time, for loading of the night trucks. This challenge only becomes more severe when more order picking is moved to other DCs, as this leads to even more shuttling of goods to DC4.

The increased goods flow results in more goods being staged at the consolidation spaces at the docks. Also, there is a lot more movement between these spaces at time of truck loading. This is because goods are staged based on their destination postal code, but can end up in different truck routes. This means that truck loaders often must move products between spaces over a long distance, which can cause congestion at the facility floor. With more goods stocked, increasing goods flow and more goods that have to be loaded, overview in the warehouse may get lost. This puts pressure on the performance and quality of the operation, with more mistakes reaching customers.

Dock doors

DC4 has around sixty dock doors, with no opportunities for expanding this number. Some years ago, there were still doors free at time of loading. Now, there are more trucks to be loaded than that there are docks. This means that maneuvering trucks between docks is necessary during the loading process. Especially for loading the night trucks this puts the process and time windows under pressure. Loading night trucks can start from 20.30PM and trucks need to depart at 22.30PM. This means there is no time for extensive switching of trucks on docks and loading trucks at one dock sequentially.

1.4.4 Sustainable transition

Sustainable transport is becoming more and more important and is already starting to become a must in some cases. Rensa underlines this importance with its goal of becoming CO2 neutral by 2030. In the transition to sustainable transport, electrification seems to become the dominant strategy for the foreseeable future. In 2021 Rensa started experiencing with electrical distribution in two pilot projects. These projects consisted of a test phase with an electrical box truck in Rensa's daily operation (Transport & Logistiek (2021), Logistics valley (2021)).

Electrical trucks have a limited range compared to a conventional truck running on diesel. This may become a problem for Rensa, performing all distribution from Doetinchem. Having decentral cross docking locations decreases the average distance to customers in end distribution. Although the range of electrical vehicles will increase in the future, having several decentral locations will ease the transition towards electrification and open up opportunities earlier.

Something that is already current, is regulation regarding distribution in city centers. Many cities already have emission zones, and a growing number of cities is moving towards zero emission for their city center, already by 2025. Having a cross docking location close to these cities may prepare Rensa for this development, making distribution by electrical truck or van easier.

1.4.5 Conclusion

Concluding, we can derive two main problems from the previous subsections:

- ***Outbound logistics warehouse DC4 lacks capacity in terms of docks doors and floor space***
- ***The current logistics network is not suitable for electrification and city center distribution***

These two problems will be leading in assessing the suitability of a new logistic network design with one or more crossdocking locations.

1.5 Research objective

Now we have indicated that this research will focus on investigating the viability of a decentral logistics network with one or more crossdocking hubs in the Netherlands, we can define our research objective.

The objective is formulated as follows:

- *Assess the viability, with a specific focus on transport costs, of locating one or more decentral crossdocking hubs in the logistics network of Rensa.*

Here, viability is determined by the transport costs resulting from a new logistics network design, the workload that shifts away from DC4 and its suitability with electrification and city center distribution.

In achieving this objective, two main research outcomes can be distinguished. First, we want to assess the impact of implementing crossdocking facilities in the outbound logistics network on the business of Rensa in a qualitive manner. Rensa wishes to gain a better understanding of the total impact that implementing a cross docking network has on the organization. We want to form a complete overview of the opportunities and challenges that emerge and the people and business units that are affected by the implementation of a new network design. Besides gaining understanding for Rensa, this part of research will give direction for the remainder of this report.

In the second part of this research, we move to quantitively analyzing the new logistics network design. Location and routing models are used to analyze the number of hubs and their location, in combination with a detailed analysis of transport costs.

Based on this research, Rensa should gain detailed understanding of the impact on transport costs that implementing a crossdocking network has. Further, this research should enhance the broader understanding of the impact implanting such a network has. This can help Rensa in indicating areas that ask for further research and in determining the follow-up steps its going to take.

1.6 Research questions

The research questions that will be central in this report are listed in Table 1 below.

Table 1 Research questions

RQ1	**What challenges and opportunities emerge when implementing decentral crossdocking warehouses in Rensa's logistics network?**
Rq1.1	*What challenges and opportunities of a crossdocking network are listed in literature?*
Rq1.2	*What challenges and opportunities identified from literature are most relevant in the problem context of Rensa?*
RQ2	**How can the investigated network with crossdocking hubs be modelled using Operations Research methods?**
Rq2.1	*How can we define the logistics network with crossdocking hubs investigated in this research?*
Rq2.2	*What Operations Research models on analyzing logistics networks are described in literature?*
RQ3	**How can the chosen Operations Research modelling techniques be implemented?**
Rq3.1	*How does the architecture behind the implemented routing machine used in our models look like?*
RQ4	**What is the impact of integrating crossdocking hubs in the logistics network on Rensa's outbound logistics?**
Rq4.1	*What number and location of crossdocking hubs form promising network configurations?*
Rq4.2	*How does the direct goods flow via Rensa's outbound logistics warehouse change under a crossdocking network?*
Rq4.3	*What are the transport costs for various numbers of crossdocking hubs?*

1.7 Scope

In this section we define the scope of this research project.

This research will only focus on the outbound logistics network of Rensa. This includes the goods flow from Rensa's central DCs to its customers. Also the goods flow between DCs of Rensa will be considered, as these goods are already ordered and on their way to the customer. Suppliers that deliver to Rensa's warehouses are outside our scope.

Further, this research will only investigate a decentral logistics network with one or more crossdocking hubs. Groters (2020), performed earlier research and concluded that this option is the most promising alternative to Rensa's current network design. This view is especially driven by the ambitious growth that Rensa strives for over the coming years. Only focusing on this design alternative makes it possible to move from broad qualitative research as performed by Groters, to a more in depth investigation of this specific network design. Operations Research techniques will help us to achieve this more in dept quantitative analysis.

To decide on the design of the future logistics network, Rensa should be able to compare the new network using hubs, with the current network design. This means we should try to achieve computations of the costs of a new network such that they can be compared to actual costs in the current network. Outcomes should resemble the real transport planning and transport costs as close as possible. Also, we should be able to map and analyze the current logistics network design as well. For this comparison we look into Operations Research models. Our modeling approach focuses on transport in a new logistics network. Focus on other topics that are not directly transport related, and cannot be covered in Operations Research models are not taken into account or discussed only briefly.

1.8 Reader's guide

The remainder of this research is structured as follows:

Chapter 2 is about context analysis, and looks into the challenges and opportunities that come with the implementation of a crossdocking network. By combining literature with input from employees of Rensa and external respondents, an answer is given on research question 1.

Chapter 3 forms the theoretical basis for our modelling approach. We define the crossdocking network that will be central in the Operations Research methods we use. Literature review is performed on location modeling, location-routing problems and the multi-trip vehicle routing problem to form an answer on research question 2.

Chapter 4 uses the insights obtained in Chapter 3 to come to an implementation of the chosen modelling techniques. Candidate locations are chosen, the model implementations are defined and the architecture behind the routing machine used in the models is described. Together, the chapter answers research question 3.

Chapter 5 analyzes the results obtained from our implemented models. Promising combinations of hub locations are given, the direct goods flow via DC4 is assessed under different solutions, and total transport costs are analyzed, so that an answer on research question 4 is obtained.

Chapter 6 concludes this research by giving the most important conclusions, discussing research limitations, giving recommendations and proposing possible leads for further research.

2. Context analysis: Challenges and opportunities in a crossdocking network

This chapter aims to form a complete picture of the problem context of implementing a crossdocking network at Rensa. The chapter gives an answer to research question 1:

RQ1 What challenges and opportunities emerge when implementing decentral crossdocking warehouses in Rensa's logistics network?

We aim to align views from different positions in the organization and combine literature with knowledge and expertise available within Rensa itself. The procedure we follow to answer research question 1 can be found in Figure 4. We perform a literature review on challenges and opportunities in crossdocking, followed by a set of interviews and a survey, leading to a list of factors that are important to Rensa.

The literature review can be found in Section 2.1. The performed interviews and survey are described in Section 2.2. The conclusions we draw are given in Section 2.3.

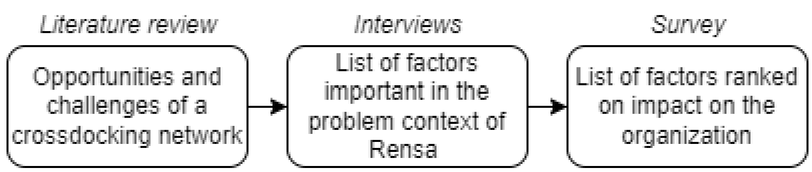


Figure 4 Context analysis structure; we combine literature and company knowledge to answer research question 1

2.1 Literature review

This section describes the literature review that forms the theoretical basis of this chapter. It answers research question 1.1:

Rq1.1 What challenges and opportunities of a crossdocking network are listed in literature?

Based on 25 articles that were considered useful, we derived a list of 18 distinct opportunities and a list of 23 challenges. These lists were created by combining all factors discovered in the different literature sources. The full list of sources with the specific factors found per source can be found in Appendix C: Literature review. To create some clarity in the long lists of factors, they are grouped into categories similar to the categorization Groters uses for performance indicators to evaluate logistics network designs (Groters, 2020). The defined categories are:

1. *Customer related*
2. *Logistics costs*
    - *Transport related*
    - *Inventory related*
    - *Warehouse related*
3. *Synchronization/information related*
4. *Other*

2.1.1 Opportunities

We start this review with the opportunities of crossdocking networks, of which the full list obtained from literature can be found in Table 2.

The general message we obtain from literature is that by implementing crossdocking hubs, one is able to increase customer satisfaction and service levels by being able to deliver faster, in smaller volumes and more frequently, while at the same time, having a possibility to reduce transport costs and decrease inventory and warehousing activities. Buakum & Wisittipanich (2019) for instance, name the benefits of reducing transportation and inventory holding costs over a traditional warehouse, while also increasing cycle-time and customer satisfaction. Buijs, Vis & Carlo (2014) name enabling consolidation of multiple less-than-truckload shipments to realize economies in transportation costs as an important purpose of a crossdock, while at the same time enhancing distribution responsiveness by the rapid transshipment of products at the facility.

Possibilities to reduce inventory and warehousing activities are the most frequently named chances in literature, named by 18 sources. Also, half of the literature sources talk about increased transport efficiency or reduced transport costs. Ahangamage et al. (2020) for instance, highlight the opportunity of goods that can be mixed or re-sorted out at a distribution center close to the customer, which will enhance the driver effectiveness. Also, they point out that crossdocking is in line with lean warehousing principles, having very few inventory levels, low transportation costs due to product consolidation and increase in product flow speeds.

Table 2 Crossdocking network: Opportunities

	Opportunities
Customer related	- Improved customer satisfaction - Improved customer service (levels)
Transport related	- Goods can be delivered faster, in smaller volumes and more frequently - Efficient consolidation of mainly LTL shipments (in- and outbound), which reduces transport costs - Reduced (noise) pollution, road accidents and urban blight
Inventory related	- Reduced or completely eliminated inventory (costs) - Reduced inventory cycle time
Warehouse related	- Reduced or completely eliminated handling (costs) (storing, picking, sorting) - Reduced labor (costs) - Reduced storage space and warehousing area - Increased handling capacity, because of reduced warehouse space needed - Final value creation can be fulfilled - Building less costly compared to warehouse or distribution center
Other	- In line with lean manufacturing principles - Accelerated cash flow due to the elimination of a storage point in the supply chain - Reduced cycle time decreases time-related risks in the supply chain and improves flexibility (damage, obsolescence, delayed product differentiation) - Improved supplier relationships - Reduced time to market

Normally in crossdocking, the crossdock location serves as a consolidation point between multiple suppliers and multiple customers. In Rensa's role as a wholesaler, its current warehouses already consolidate the streams from the supplier side. Adding a crossdocking hub in Rensa's logistics network will therefore not lead to the decrease in stock as described in literature. Improvement for Rensa can be on the transport side. Trucks do start delivery tours from Doetinchem, but a consolidated goods stream is sent to a hub. Sorting on separate delivery tours happens there, on a location closer to the end customer.

2.1.2 Challenges

We see that literature mainly focusses on overcoming the challenges that occur in implementing a crossdocking network. This was also observed by Buakum & Wisittipanich (2019) who see that cross-docking raises numerous questions with regard to optimization so that more research studies are conducted to improve and make the operations in crossdocking more efficient and effective. This shows in a longer list of challenges, compared to the opportunities as can be seen in Table 3. We see that the list of challenges is more diverse and that there is less overlap between different sources.

Under the customer related grouping, we see several challenges that actually read more as prerequisites for a successful crossdocking network:

- A predictable, high volume, constant demand rate is preferred (Naunthong & Sud-On (2020), Braglia, Castellano, Gallo, & Romagnoli (2019), Ahangamage et al. (2020))
- Longer shelf life and low value products are most suitable (Benrqya, 2019)
- More suitable in a situation with many stores or customers to serve (Braglia, Castellano, Gallo, & Romagnoli, 2019)
- More suitable when truck loads deviate significantly from full-truck-load, which should be a primary focus (Galbreth, Hill, & Handley, 2008)

Under warehouse related, we find some challenges about the complexity of the organization and operations at the crossdocking hub. Buakum & Wisittipanich (2019) for instance, talk about the complexity of the task scheduling problem between in- and outbound dock doors. Other relevant challenges under this category are:

- Additional stop for consolidation causes double handling activities and excessive move time (stocking, depalletization), this could potentially increase material handling costs and labor (Stephan & Boysen, 2011)
- Many shipments and fast transshipment process may lead to an increased risk of errors (Bilinska-Reformat & Dewalska-Opitek, 2013)
- Reduced flexibility in work balancing and minimization (Schaffer, 1998)

We see that factors regarding the importance of coordination, synchronization and information flow throughout the supply chain are listed the most among all literature sources. This shows that this experienced as a very important challenge in crossdocking. Buakum & Wisittipanich (2019) and Buijs, Vis & Carlo (2014) for instance, both stress the importance of the synchronization of both local and network wide operations. Buijs, Vis & Carlo (2014) therefore advise a holistic approach in the design and coordination of a crossdocking network.

Both Ahangamage et al. (2020) and Van Belle, Valckenaers, & Cattrysse (2012) stress the increased importance of effective information flow in a crossdocking network, compared with regular distribution. Further, it is pointed out by Buijs et al. (2012) that a crossdocking network is highly sensitive to dynamics in the environment, more than a standard warehouse. This can lead to different types of waste, such as temporary storage or unevenness in workforce allocation. An advantage for Rensa is that the central DCs are the suppliers of crossdocking locations, so there is no need for synchronizing operations with many different third parties.

Table 3 Crossdocking network: Challenges

	Challenges
Customer related	- Medium to high, constant demand is preferred as demand fluctuation can significantly disturb crossdocking operations - Longer shelf life and low value products are most suitable - More suitable in a situation with many stores/customers to serve - More suitable when truck loads deviate significantly from FTL, this should be a primary consideration - Only if the customers accept the longer delivery times will the additional processing time for double handling in the cross-docking terminal (compared with a point-to-point delivery) seem acceptable
Transport related	- Suppliers should be reliable and quick in terms of response, quantity and quality of products received - Significant skills are needed to coordinate the many suppliers, the various inbound materials, and the demand of the various customers that a cross dock serves
Inventory related	- Higher chance of stock out due to decreased inventory, some safety stock or hybrid system is advised
Warehouse related	- Requires (diversely)skilled management and staff, additional training may be needed - A high number of vehicles is needed to always be ready for frequent and large shipment deliveries - Additional stop for consolidation causes double handling activities and excessive move time (stocking, depalletization), this could potentially increase material handling costs and labor - Total distance traveled and the congestions that appear on-route between dock doors are important factors regarding the operational performance of a cross-dock - The operational performance at cross docks is greatly influenced by the real-time decisions made by the dispatchers in order to control on-going transshipment operations - Task scheduling between inbound and outbound dock doors of cross-docking is a complex scheduling problem in itself since multiple resources need to be coordinated and each worker needs to be scheduled in detail - Someone responsible for tactical management is of vital importance (i.e. door allocation, resources, should be separate from first line supervisor) - Reduced flexibility in work balancing and minimization
Synchronization/ Information related	- There is no storage buffer, so local(within) and network-wide (across supply chain partners), in- and outbound operations need to be highly synchronized and coordinated, this requires a holistic approach - Information flow, both local(within) and network-wide (across supply chain partners) is significantly more important, it is for instance necessary to have an overview of the workload to plan the upcoming days - There is a need for tailored information technology (with order management, advanced shipping notice, yard management system, cross dock management system, track and trace across the supply chain, suitable WMS) simplifying the procedures for orders placement and execution, this comes with increased costs - Sophisticated technological systems and processes are needed to optimize transportation routes, maximize the volume of material on trucks, adapt to unplanned events, and ensure that material flows efficiently through the cross dock in a manner that maximizes quality, minimizes cost, and is timely - The operational performance at cross docks is highly sensitive to dynamics in the environment, more than in a standard warehouse (arrival times and number of trucks, waiting times for drivers, varying processing times, resource availability), this creates different types of wastes such as

	temporary storage, unevenness in workforce allocation and quality issues.
Other	- Many shipments and fast transshipment process may lead to an increased risk of errors, where the time and space to correct significant errors is not available - High investment costs on both infrastructure and technologies

2.2 Interviews & survey

This section describes interviews and a survey, following on the literature review executed in the previous section. We aim to answer research question 1.2:

Rq1.2 What challenges and opportunities identified from literature are most relevant in the problem context of Rensa?

We start by describing the goal, participants and procedure of the interviews and survey. After that, we discuss the obtained results, that lead to the conclusions given in Section 2.3.

2.2.1 Goal

Results from literature are often quite abstract and not all factors listed in Table 2 and Table 3 are relevant for this research. Interviews are carried out to make the translation from literature to specific factors that are relevant in the problem context of Rensa. Goal of the interviews is to come to a list of factors that together give a complete as possible overview of the impact of implementing a crossdocking network at Rensa. The survey following on the interviews has the goal to rank the factors based on their impact, by bundling the viewpoints of different departments of Rensa.

2.2.2 Participants and procedure

Interviews and survey are undertaken with respondents from within and outside Rensa. Six employees of within Rensa are interviewed. They are each from different departments, to represent a wide array of functions and viewpoints concerning a new logistics network design.

Three external respondents complement the internal interviewees. Each of the organizations they are from have their logistics organized differently. We have one logistics service provider that operates a network with six hubs divided over the Netherlands and Belgium. Their hubs have both crossdocking and storage functions. One company uses a logistics service provider for its complete logistics and one company performs all transport internally. This last company processes fresh products, and is therefore dependent on just in time deliveries and crossdocking operations. Their logistics network evolved more decentral over the last years. All these different viewpoints may add valuable insights to the outcomes of the respondents of Rensa. The list of respondents can be found in Table 13 in Appendix D: Interviews.

The interviews are semi-structured. We use the literature review as basis for the interviews, and not a list of complete beforehand prepared questions. The researcher has the task to follow the factors from the literature review and make sure that all factors are touched upon. This is done because we want the interviews to be open conversations, more than being a question & answer session. This should invite respondents to think further about the context of the proposed opportunities and challenges so that a complete set of relevant factors for Rensa is obtained in the end.

The survey aims to rank the factors specifically based on their impact on the problem context of Rensa. Therefore it is only submitted to the respondents from within Rensa. The internal respondents are asked the following question:

- *Assess the 19 identified factors from the interviews, based on their impact in the problem context of Rensa.*

Here impact can be seen in the broadest context possible. It ranges from financial impact to impact on customers and the people and processes within Rensa. Respondents can score each factor on a scale from 1 to 10. A score of 1 means that a factor has no impact in the research context of Rensa. A score of 10 means that a factor has high impact and is essential in the research context.

2.2.3 Results

This subsection gives the results of both the interviews and adjacent survey.

By combining the interview outcomes of all respondents, 19 distinct factors are obtained. The 19 factors are ranked in order of the survey results for all respondents, leading to the list of factors in Table 4 below. In Table 14 in Appendix D: Interviews the interview outcomes are listed per respondent, followed by a description of each factor.

As we saw in literature, respondents of our interviews are more focused on the challenges that we have to overcome than on the opportunities that may arise when changing the logistics network. However, the factor emphasized by almost all respondents and considered most important is the opportunity of using trucks more flexibly and efficiently from hubs. All respondents are interested in the opportunities of organizing transport from hubs more efficiently. Especially the opportunity of driving multiple delivery trips, with average distances to customers getting smaller from a hub, is encouraged by several respondents.

The next four highest scoring factors are all challenges, with the effect of hub size on the effectivity and efficiency of the organization on site being regarded the most important. Several respondent stress that it may be difficult to set up a stable organization when a hub is small and remote from the central organization.

Two factors that are named in one breath by several respondents are the effect on transport planning and a more variable demand pattern on a hub due to smaller scale. Both are related to the fact that we are dividing the big pool of demand into smaller sub parts. This may affect the efficiency of Rensa's transport planning and cause increased complexity for Rensa's planners. Having severe demand fluctuation at hubs, asks for an organization and process that is able to serve these fluctuations.

The factor complementing the top five factors regarded most important is about the chance of errors in the logistics network. Part of the products go through an extra handling step in a new network design using hubs. Extra loading, unloading and movement of goods increases the chance of errors. In Rensa's central DCs a product damaged during loading can quickly be replaced from stock, something that is not possible on a hub anymore. What may reduce this risk is that handling at a hub takes place on smaller scale and is less complex compared to the operation at the central DCs.

Table 4 Interview outcomes in order of average survey score

Rank	Factor	Survey score
1	More flexible and efficient use of trucks from hub	9.00
2	Effect of hub capacity on the effectivity and efficiency of an organization on site	8.33
3	Effect on transport planning	7.67
4	More variable demand pattern on hub due to smaller scale	7.50
5	Chance of error	7.50
6	Deliver project inventory from hub	7.33
7	Network suitable for in-house electrical (urban) distribution	7.33
8	Handling	7.33
9	Impact of a more decentralized network on information flows and IT	7.17
10	Night distribution via hubs	7.00
11	Truck fleet composition and delivery	6.83
12	(Partially) combining counter function and hub	6.00
13	Return flow through hubs	6.00
14	Construction site logistics	5.83
15	Put the sales order, awaiting to be called off, already on the hub	5.67
16	Effect on people in existing organization	5.33
17	Same day delivery	4.83
18	Return of waste from customers	4.50
19	Direct delivery from regional supplier to hub nearby	3.83

2.3 Conclusion

This chapter combines literature with interviews followed by a survey to gain understanding of the factors that are relevant for Rensa when a new logistics network design is considered.

Literature points out that it is possible to increase customer satisfaction and service levels while at the same time reducing inventory and warehousing activities. We have seen that in Rensa's network, a crossdocking hub will not save a storage point in the supply chain and therefore not lead to the decrease in stock as described in literature. Improvement for Rensa can be on the transport side as also showed in literature. This gives reason for this research to have main focus on transport costs and related concerns.

Interviews gave us a list of 19 distinct factors that may be relevant to Rensa when regarding crossdocking in the logistics network. The factor considered most important by respondents is the opportunity of using trucks more flexibly and efficiently from hubs. Achieving more efficient transport from a hub will be of central focus in this research. Two other factors are also transport related and will come back in the remainder of this research. These are the effect on transport planning and a more variable demand pattern on a hub due to smaller scale.

Two other factors among the five factors regarded most important by respondents of the interviews and survey are not directly related to transport. The effect of hub size on the effectivity and efficiency of the organization on site and the chance of errors in the network are beyond our transport focus and could be separate research topics on their own. Modeling Rensa's logistics network will gain understanding of the size that hubs will have, gaining some understanding of the size of an organization on site. However, no further research is performed on this topic.

Factors that are regarded of less importance from our survey results will not be focus of this research. Also, most of these factors are beyond the scope of this research as they are not centrally about transport and more operational of nature.

Network suitable for in-house electrical (urban) distribution is a factor that is already named in the problem identification of this research and certainly forms and opportunity when moving to a new logistics network. Average distances to customers following from a modeling approach will give an identification on the ability of performing electrical distribution. However, this factor is also operational of nature and the topic asks for research beyond transport considerations from our modeling approach.

The factor night distribution via hubs may be an option when Rensa decides to implement a network with crossdocking hubs. However, together with Rensa it is decided to first focus on day distribution to show the potential of the proposed logistics network design. Night distribution can be an opportunity that is considered later in the process of investigating and implementing hubs.

3. Finding a suitable modeling approach

In this chapter we explore literature, to find a modeling approach that is able to calculate the impact on transport costs of implementing one or more crossdocking facilities in the logistics network of Rensa. An answer is given to research question 2:

RQ2 How can the investigated network with crossdocking hubs be modelled using Operations Research methods?

The first three sections of this chapter focus on defining the properties of the problem that we are solving with our modeling approach. Section 3.1 describes the proposed logistics network, that is central in this research. Section 3.2 defines the scope of our modeling problem and clarifies the modeling task that lies ahead by placing our problem on the strategical, tactical, operational horizon. In Section 3.4 we look into the properties of different location modeling formulations.

Next sections 3.5 and 3.6 look into location-routing problems, and multi trip location problems and are more focused on solution methods. Although literature sections, we end each of the sections 3.4, 3.5 and 3.6 with a short conclusion, with the goal to gain a logical line or reasoning throughout the chapter.

3.1 Defining the proposed logistics network

This section defines the logistics network investigated in this research. A textual description of the network is given in Section 3.1.1, where Section 3.1.2 defines the network using graph theory. The research question that is answered in this section is research question 2.1:

Rq2.1 How can we define the logistics network with crossdocking hubs investigated in this research?

3.1.1 Proposed network description

In the proposed logistics network design of Rensa, one or more hubs that operate according to a crossdocking principle are located throughout the Netherlands.

As already defined in Section 1.3, in crossdocking we transfer incoming shipments directly to outgoing vehicles without storing them in between (Van Belle, Valckenaers, & Cattrysse, 2012). All shipments leave the terminal within 24 hours(e.g. Konrad & Boysen (2011), Van Belle, Valckenaers & Cattrysse (2012), Buakum & Wisittipanich (2019)). At the crossdocking hubs shipments are split and sorted into individual routes for customer delivery. Smaller trucks of type city trailer and box truck, together with delivery vans, deliver the orders to the customers from the hub.

The crossdocking hub receives large shipments from the central DCs in Doetinchem. This shuttling traffic is performed using so called LHVs (longer heavier vehicle), aiming at maximum capacity utilization. There is no goods flow between the hubs mutually. The hubs do not hold stock and are directly supplied from individual DCs as much as possible, without consolidating at DC4. This means that we aim to make all transport from central DCs to hubs point-to-point, direct delivery.

In contrast to the current truck pitches, the hub is a point where goods can be consolidated and trucks are loaded. This opens the possibility to reload trucks and do a second delivery trip in order to maximize loading capacity and driver working time utilization. Under the new network, the truck pitches that are currently located throughout the Netherlands will close. Their function will be replaced by the nearest

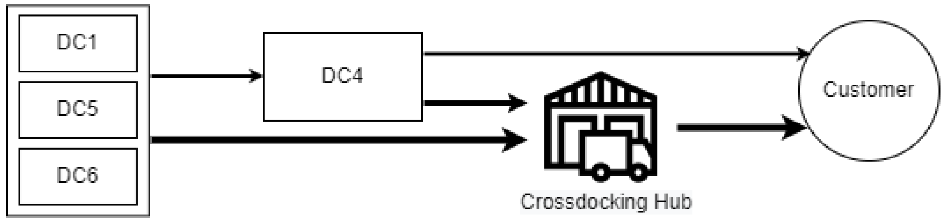


Figure 5 Proposed logistics network design

hub. As locating hubs will most likely be a gradual process, also the closing of truck pitches will not happen all at sudden.

A schematic representation of the new logistics network design can be found in Figure 5.

3.1.2 Graph representation

A logistics network can be presented as a graph G(V,E). The graph is composed of nodes(or vertices) V and edges E. The nodes in the graph represent points where goods flow to or from. The edges in the graph connect the nodes.

The graph representing the logistics network of Rensa can be found in Figure 6 below. The nodes(or vertices) V represent the set of Rensa's DCs, the set of hubs and the set of customers. Edges E represent the roads along where goods are transported. The graph is undirected as edges can be traversed both ways. Also, the direction of travel does not influence travel duration and transport costs. The graph only shows the goods flow via hubs. The fraction of goods flow that is still distributed directly from DC4 is ignored for clarity and because this part of distribution does not change from the current situation.

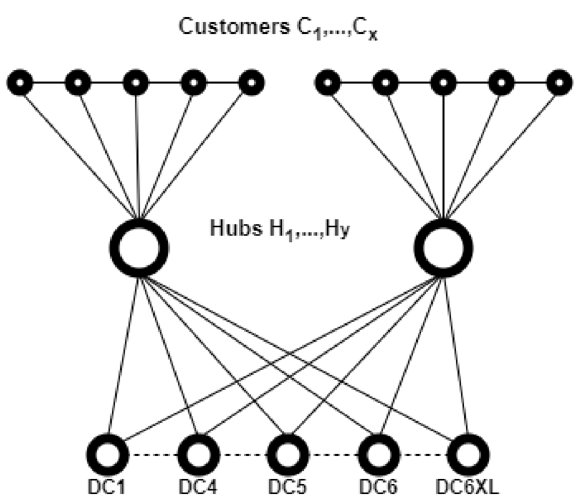


Figure 6 Graph representation of the proposed logistics network of Rensa

The four DCs of Rensa are connected to the hubs via a point-to-point connection. This means that no traffic occurs between DCs and between hubs mutually. This is an ideal situation for the main process, but we are aware that for capacity utilization it may be necessary to combine streams from different DCs. This will at least be the case for the last time shuttling to a hub, when at all DCs the last picked orders have assembled. The edges between DCs and hubs are traversed by trucks of type LHV only.

At the hub nodes, the truck type that transports goods changes. This means that the edges connecting hubs and customers are traversed by box trucks, city trailers and delivery vans. Customers are visited in routes, instead of a point-to-point connection. This means that there are also edges connecting the customers mutually. Each customer is always served from the same hub, which means that there is only one edge between a customer and one hub. Each hub has a dedicated truck fleet assigned to it. Delivering vehicles do not visit other hubs, which means there are no edges connecting the hubs. Under the new logistics network it will be allowed for trucks to reload at the hub and do a second delivery trip. This means that hub nodes may be visited twice.

3.2 Scope of modeling approach

This section places the problem of choosing the number and location of hubs and calculating transport costs on the horizon of operational, tactical and strategical problems.

The location problem in which one ore more facilities are to be located in a given area is a clear strategical problem, and a critical aspect of strategic planning for a broad spectrum of public and private firms. A facility location problem is strategic of nature as the development and acquisition of a new facility is typically a costly, time-sensitive project. Before a facility can be purchased or constructed, good locations must be identified, appropriate facility capacity specifications must be determined, and large amounts of capital must be allocated (Owen & Daskin, 1998). Even when Rensa decides to outsource the activities on a hub, they still strive for a long term agreement for minimum three years. Also then, investments will need to be made in adapting processes, IT and fleet.

Where the location of facilities forms a strategical decision on itself, it clearly has a relation with the operational problem of vehicle routing. In this research, besides solving a location problem and allocating customers to locations, we also want to give a detailed comparison on transport costs between the current logistics network of Rensa and a new situation. To calculate transport costs we need to know the number of trucks deployed and driving times and distances on specific days. For this, we need to solve vehicle routing problems. What forms an extra incentive for vehicle routing, is that we want to investigate the option of allowing for multiple delivery trips from the hubs. The literature review in the next section will investigate the field of vehicle routing problems with multiple trips (VRPMT).

We can solve the location and vehicle routing problems separately. An interesting question however, may be if it has benefits in the problem context of Rensa to incorporate routing in the location problem. Then we are dealing with the specific branch of locational analysis that is location-routing. We can define location-routing as: ‘location planning with tour planning aspects taken into account’ (Bruns, 1998). Besides focusing on location problems and vehicle routing, the literature review following in the next section will also investigate the field of location-routing. This way we want to investigate if location-routing is valuable in this research.

3.3 High-level positioning of the modeling approach

The modeling approach we take in this research on integrating crossdocking hubs in the logistics network of Rensa will focus on deterministic models, with a static planning period and heuristic solution methods.

Rensa’s daily operation provides us with plenty of deterministic input data. Demand fluctuations that may influence the success of a crossdocking network are already present in this data. Stochasticity is a relevant daily challenge for Rensa. However, flexible capacity is used to cope with uncertainty and it has no effect on the general network characteristics as the location decision and routing problem.

We focus on the class of models with a static planning period. The primary focus of our research, is to see if the current operation of Rensa could benefit from a logistics network using crossdocking hubs. This kind of snapshot approach fits a static planning period. Future growth however, can influence the location decision and size of crossdocking locations. When this needs to be addressed, this can be done by changing input data, rather than taking a dynamic modeling approach.

Solving implemented models will be done using heuristic solution methods. We investigate the classes of location and routing models, problems that are both NP-hard of nature. Small problem instances may be solved to optimality. However, this is not a possibility for the large scale real life problem we face at Rensa.

3.4 Literature review: Location modeling

In this section we review literature to form an answer on research question 2.2:

Rq2.2 What Operations Research models on analyzing logistics networks are described in literature?

After this section we should be able to define a modeling approach that suits analysis of the proposed logistics network. Section 3.4 starts with a general review of the field of location modeling. After that we zoom in on the option of incorporating the operational routing problem in the location decision. We do that by investigating the field of location-routing problems in Section 3.5. We conclude the review with routing considerations in Section 3.6, where we specifically investigate multi trip location routing problems.

Location analysis is about the modeling, formulation and solution of a class of problems that can best be described as siting facilities in some given space. Location problems are characterized by four components: (1) customers, who are presumed to be already located at points or on routes, (2) facilities that will be located, (3) a space in which customers and facilities are located, and (4) a metric that indicates distances or times between customers and facilities (ReVelle & Eiselt, 2005).

3.4.1 Taxonomy

Daskin (2008), provides a clear taxonomy of location models, based on the space in which the problems are modelled. Four branches of location models can be distinguished. These are **(1)** analytical models, **(2)** continuous models, **(3)** network models and **(4)** discrete models. For clarification we added a visual representation of the models described by Daskin (2008) in Figure 7. The next paragraphs describe each branch of models separately.

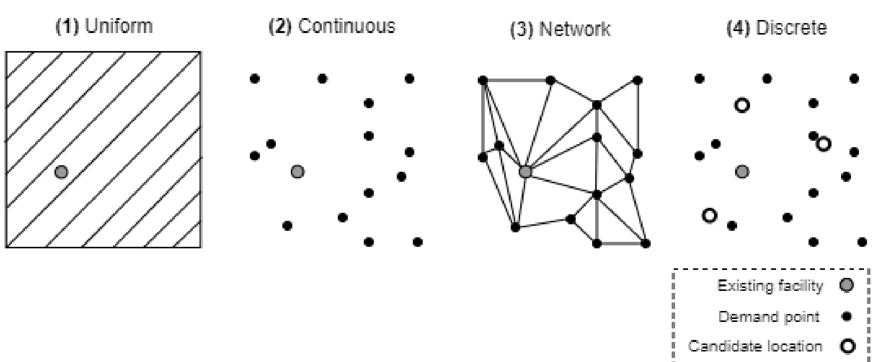


Figure 7 Taxonomy of location models, schematic overview. Based on Daskin (2008)

In **(1)** analytical models, it is assumed that the demand is distributed in some way over a continuous service area. Demands can for instance be distributed uniformly over a square area *a*. Facilities can be located anywhere within that area *a*. Analytical models are typically solved using calculus or other simple techniques. This makes analytical models the simplest of location models, but the strong assumptions limit their practical applicability. However, analytical models may still be useful, as is advocated by Daskin (2008).

Traditionally logistical problems are solved by gathering as much detailed information as possible about the problem upfront. This often means onerous data collection effort and NP-hard optimization problems. These need simplification and heuristic solutions to be solved, not necessarily leading to solutions close to optimal. In the method by Daskin (2008), detailed data are replaced by concise summaries and numerical methods are replaced by analytical models upfront. These simplifications make it possible to solve accurately, so that one can get acquainted with the properties of solutions close to the global optimum. These near-optimal solutions can then be used to formulate guidelines for solutions that satisfy all the detailed requirements ignored in the simplified analysis. Another advantage of the simplified analysis, is that qualitative insight can be developed(e.g. into the most important trade-offs influencing the choice of final solutions) and communicated meaningfully to managers and decision makers due to the lack of interfering details.

(2) In continuous models demand still occurs on a continuous service area. However, the demand itself occurs at specific points that are defined by coordinates which may change continuously. Facilities can still be located anywhere on that same continuous service area. The Weber problem, in which a single facility is to be located to minimize the demand-weighted total distance, is a typical example of a continuous model.

(3) Network models assume that demand arises, and facilities can be located, only on a network composed of nodes and links. Often demand occurs only on the nodes, while facilities can be located anywhere, both on nodes and links. In network models, besides where facilities should be located, one must also determine which arcs should be included in the network. Typical examples are the design of subway or rail systems, electricity distribution systems, and computer networks (Current, Daskin, & Schilling, 2001). Most literature on network models is focused on finding polynomial time algorithms, often for problems on specially structured networks such as trees.

The last branch of location models consists of **(4)** discrete models. In discrete models facilities locations are restricted to a finite set of candidate locations. This distinguishes discrete models from continuous models, in which facilities could be located anywhere in the service region. It makes discrete models the most realistic location models, but computational and data collection costs are high (Goetschalckx). In discrete models there may or may not be an underlying distance metric. Distances or costs between any pair of nodes may be arbitrary, but generally do follow some rule. Possible distance rules are Euclidean, Manhattan, network, or great circle distance.

Interpretation

In our comparison between the proposed and current logistics network of Rensa, we want to stay as close to reality as possible. This makes analytical models, with many simplifications done beforehand, less suitable in our problem context. Also network models, where specific focus is on which links to open or construct, is no good fit with our research. As trucks can drive the complete road network of the

Netherlands, we have no concerns about choosing or opening specific links as is the case in for instance a subway system.

Continuous and discrete location models both model customer demand as discrete points, which suits our situation. There are some specific locations or conditions to locations given by Rensa, that make it valuable to be able to model specific hub locations. This makes us decide to zoom in further on the class of discrete location models in Subsection 3.4.2.

3.4.2 Discrete models

This subsection describes the different branches of discrete location models. Daskin (2008) divides discrete location models into **(1)** covering-based models, **(2)** median-based models and **(3)** other models. A breakdown of the branch of discrete location models can be found in Figure 8 below.

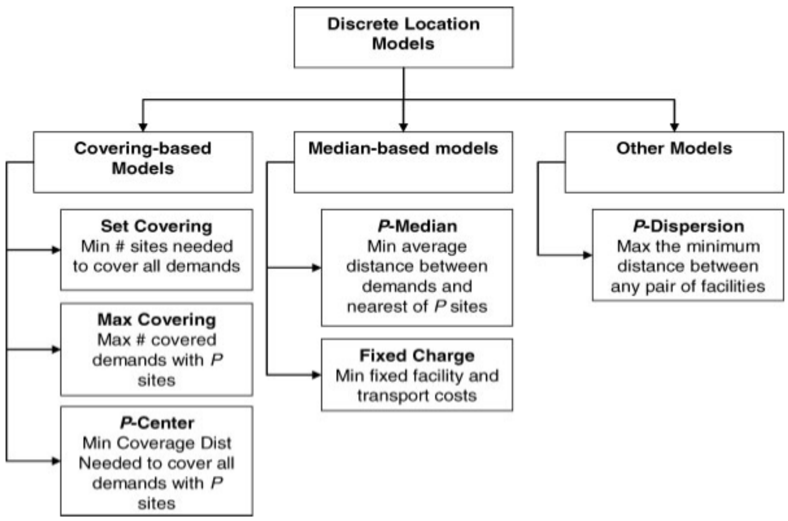


Figure 8 Discrete location models (Daskin,2008)

(1) Covering-based models assume that there is some critical coverage distance or time within which demands need to be served to be counted as "covered". Typical applications of these models are in designing emergency services where there are often both practical and legislative guidelines for coverage. In the set covering problem, the objective function minimizes the number of facilities needed to cover all demand points. The max covering problem on its turn, starts with a fixed number of facilities to locate. The objective is to maximize the number of covered demand points under the given number of facilities. Finally, the p-center model finds the smallest possible coverage distance, the distance from facility to demand point, to cover al demands under a given number of facilities.

(2) Median-based models minimize the demand-weighted average distance between a demand node and the facility to which it is assigned. Typical applications of these models are in distribution planning contexts in which minimizing the total outbound or inbound transport costs is essential. Two examples of median-based models are the p-median model and the uncapacitated fixed charge location problem.

In the p-median model, p facilities are located to minimize the demand-weighted total distance between demands and the nearest facility. The p-median model is NP-hard and heuristic solution approaches are generally proposed for practical problems. Where the p-median problem ignores differences in the facility location costs at different sites, the uncapacitated fixed charge location problem accounts for these costs in the objective function. The problem has the same constraints as the p-median problem with the exception that it does not specify the number of facilities upfront.

Daskin (2008) classifies one model in the category **(3)** other models. This is the p-dispersion model, in which the minimum distance between any pair of facilities is maximized. The goal here is thus to maximize instead of minimize distances between facilities. This problem finds its application for instance in locating franchise outlets, where cannibalization between sites is an important factor.

Interpretation

As we look at the three branches of discrete location models, the median-based models clearly provide the best fit with the problem context of this research. In p-dispersion models the minimal distance between facilities is maximized, which has nothing to do with locating crossdocking hubs. Also covering-based problems are not a reasonable branch for this research. Counting demand points as covered or not would not be logical for the case of Rensa, as all demands are always fulfilled regardless their location. Also, it may be perfectly fine that demand points are located far from a facility if that turns out to be the best option for the efficiency of the logistics network as a whole.

What is lacking in the section on location models above, is the routing aspect to which location models and our problem context are clearly related. As we want to give a detailed comparison on transport costs between the current logistics network of Rensa and a new situation, we need to know the number of trucks deployed with their driving times and distances on specific days. For this, we need to solve vehicle routing problems. We can solve the location and vehicle routing problems separately. An interesting question however, may be if it has benefits in the problem context of Rensa to incorporate routing in the location problem. This the specific branch of locational analysis that is location-routing, is described in the next section.

3.5 Literature review: Location-Routing problems

This chapter describes a literature review in the field of location-routing problems. The review is based on a comprehensive survey by Nagy & Salhi (2007).

3.5.1 Definition and classification

Nagy & Salhi (2007) define location-routing from a hierarchical viewpoint. The aim is to solve a facility location problem (the “master problem”), but in order to achieve this, simultaneously a vehicle routing problem needs to be solved (the “subproblem”). Whereas basic location models assume that demand is served directly from a facility, in location-routing models we also deal with tour planning, thus having multiple stops on routes.

Location-routing problems involve three inter-related, fundamental decisions: where to locate the facilities, how to allocate customers to facilities, and how to route the vehicles to serve customers (Perl & Daskin, 1985). To classify as a location-routing problem, both location and routing aspects must be captured in an integrated solution approach. Without addressing the inter-relation between the two, one cannot speak of a location-routing problem (Nagy & Salhi, 2007).

Locations models also have an implicit metric for serving customers, be it in a direct connection between facility and customer. It becomes valuable to look into location-routing problems over basic facility location models when the location of facilities influences the performance of the routes that can be constructed. Current, Daskin and Schilling (2001) state that modeling distribution cost as the cost of a

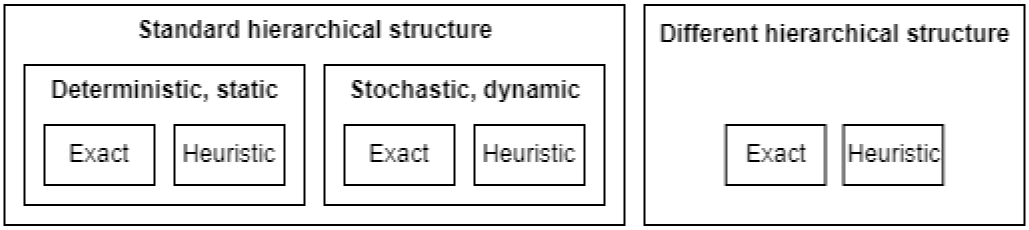


Figure 9 Structure of the literature review by Nagy & Salhi (2007)

simple round trip from one facility to one customer may significantly misrepresent the actual costs and may, as a consequence, result in the selection of sub-optimal facility sites when multi-stop tours are used.

Nagy & Salhi (2007) structure their review as shown in Figure 9. The classification looks at four key aspects of location-routing problems, being:

- *Hierarchical structure (regarding facilities, customers and the relationship between them)*
  The hierarchical structure in most location-routing problems consists of facilities serving a number of customers, connected to their depot by means of vehicle tours. When this is the case, and facilities are not connected with each other, Nagy & Salhi (2007) speak of standard hierarchical structure. Under different hierarchical structure, problems can be considered that have tour planning between facilities, or that do not have tour planning at all. The problem context of Rensa fits the standard hierarchical structure with hubs that do not serve each other, and tours connecting the customers to the hubs.
- *Type of input data, deterministic/stochastic*
  We directly use pre defined periods of historical data in our modeling approach. This makes our input data deterministic and we will focus on this class of problems.
- *Planning period, static/dynamic*
  The planning period may either be single-period or multi-period, also called static and dynamic respectively. We are interested in the class of static problem formulations.
- *Solution method, exact/heuristic*
  We will focus on heuristic problem approaches. Nagy & Salhi (2007) conclude that exact methods provide significant insights into problems, but due to the complexity of location-routing they can only tackle relatively small instances. With both the location and routing problems being NP-hard and the realistic large problem instance for which we are solving, it is unrealistic to use an exact approach.

Summarizing, we are interested in the set of problem formulations that has a standard hierarchical structure, deterministic input data, is static and which is solved using heuristic solution methods.

3.5.2 Solution methods

This section describes insights from several relevant research examples in the review of Nagy & Salhi (2007). Deterministic problems are further divided in the review, based on the way they model the relationship between the locational and the routing subproblems. This can either be clustering-based, iterative or hierarchical.

Clustering-based methods

In clustering-based methods, the customer set is divided into clusters, either per potential depot or per vehicle route.

Bednar & Strohmeier (1979) start their research by clustering customers into one, two or three clusters, based on Euclidean distances weighted with customer demands. To do this they use a k-means method. Based on the clusters' centroids they check if candidate locations determined by the customer are in favorable locations. Next, a mixed integer programming model is formulated in which the optimal locations to be opened are determined and the decline of variable transport costs is compared with the costs of an extra location. The model uses direct deliveries and Euclidean distances. The last part of the research calculates vehicle routes using the savings method. This method uses a matrix for all points and partial routes that stores the possible savings of putting points a and b are in a route together, instead of delivering them separately.

Srivastava & Benton (1990) investigate several significant environmental factors that influence distribution system design, such as ratio of location to routing cost and spatial distribution of customers. The problem is modeled as a mixed integer programming, but solving the problem exactly is not feasible. The research develops three alternative heuristic approaches. (1) A savings-drop heuristic, in which all depots are open initially, then simultaneously depots are dropped and customers are assigned to routes from open depots. (2) A savings-add heuristic works the other way around, and starts with all depots closed and opens them one-by-one. (3) The last approach is a cluster-routing heuristic, in which a desired number of clusters is defined and an equal number of depots is located on the locations closest to the clusters' centroid. The savings-drop and cluster-routing heuristics give the highest improvement over a sequential approach to location-routing problems.

Barreto et al. (2007) integrate several hierarchical and non-hierarchical clustering techniques with several proximity functions in a sequential heuristic algorithm. The heuristic consists of four steps. In step one customers are clustered with a vehicle capacity limit. Step two determines a distribution route within each cluster. The routes are improved using a 3-optimal local search procedure in step 3. Finally, the DCs are located and the routes are assigned to them by solving the single source capacitated location problem. In the clustering of step 1, four grouping methods are considered and six different proximity measures are evaluated per grouping method. The research concludes that different proximity measures perform best for different instances. Using several proximity measures and choosing the best solution is advised. The grouping methods performed well in general and the average gap between them is narrow.

Iterative methods

Iterative heuristics decompose the location-routing problem in its two subproblems. They iteratively solve the location and routing problems, feeding information from one phase to the other.

Perl & Daskin (1985) formulate the warehouse location-routing problem as a mixed integer program, but solve it using a heuristic approach based on decomposing the problem into three sequential subproblems, the complete multi-depot vehicle-dispatch problem (1), the warehouse location-allocation problem (2) and the multi-depot routing-allocation problem (3). The aim is to minimize the sum of transportation and warehousing costs.

The complete multi-depot vehicle-dispatch problem (1) assumes that all potential DC sites are used and constructs an initial set of routes to minimize the total delivery cost. The warehouse location-allocation problem (2), locates the warehouses and allocates the routes constructed in phase one. The multi-depot routing-allocation problem (3) in which simultaneously reallocates customers to warehouses and solves the multi-depot routing problem for the warehouses selected in phase two. A case study demonstrates that the model can solve large scale problems. The method of Perl & Daskin (1985) is improved by Hansen et al. (1994) with a focus on efficiency and practical useability.

Both Salhi & Fraser (1996) and Wu, Low & Bai (2002) consider a heterogenous fleet type and use the length of tours to determine variable costs. The latter decompose the problem into two sub problems, the location-allocation and vehicle routing problem. A combined tabu search and simulated annealing framework is used to solve the problem. For a problem with 55 nodes, it is found that their method outperforms both Perl & Daskin (1985) and Hansen et al. (1994). For 85 nodes the method of Hansen et al. (1994) results in the lowest costs. Salhi & Fraser (1996) use two heuristics, one for the location problem and the other for the multi depot routing problem. Problems are solved, ranging in size from 55 to 199 customers, 1 to 5 depots and 1 to 3 different vehicle capacities.

Salhi & Nagy (2009) give a non-linear programming formulation and use a heuristic to solve it. Initial solutions are found first for the locational subproblem and after that for the resulting routing subproblem. After that the heuristic takes the end points of the routes to improve the location of each depot, solving multiple Weber problems. The approach returns to solving the vehicle routing problem and stops when there is no change in depot locations from an iteration to the next anymore. Routing is based on a savings method. The location problem is solved using the Weiszfeld procedure, which finds the point that minimizes the sum of weighted Euclidean distances to a given number of fixed points.

Hierarchical methods

In hierarchical methods, the main algorithm is devoted to solving the location problem. It refers to a sub routine that solves the routing problem.

Nagy & Salhi (1996) describe their hierarchical approach as a ‘nested method’, consisting of a local search locational algorithm that refers to a routing method when evaluating neighboring solutions. The solution space consists of all possible combinations of customer sites, as all customer sites are possible depot sites. The neighboring structure is defined by the moves add, drop and shift, respectively opening, closing or simultaneously opening and closing a depot. A variant of tabu search is used to make this decision in the algorithm. The algorithm starts with an initial solution and with solving the resulting multi-depot routing problem. After that, it iterates through the location decision until no better solution is found in a certain number of iterations. The research observes that a change in location is limited to a certain ‘region’ and therefore recalculates the routes for a limited area only.

Tuzun & Burke (1999) use a similar approach in the form of a two-phase tabu search algorithm. In the first phase, tabu search is used to achieve a good configuration of facilities. For each of these configurations, another tabu search is used in the second phase, to obtain a good routing for the given configuration. The two searches may seem sequential, but they are coordinated in the sense that each time a move is performed on the location phase, the routing phase is started to update the routing according to the new configuration. In line with Nagy & Salhi (1996), the routing phase is a localized search, instead of a global exploration of all routing moves, eliminating a lot of unnecessary computation.

Lin, Chow & Chen (2002) divide their problem into three phases: facility location, routing and loading. Their method allows for multiple vehicle trips to be made and restrictions on vehicle and facility capacity and trip time limits exist. The algorithm uses heuristic and meta-heuristic approaches as local search, threshold accepting and simulated annealing, but also uses an exact approach in the form of a traveling salesman problem algorithm. As the number of nodes in a route is not very large, this exact algorithm can still run fast although the problem is NP-hard. Threshold accepting was introduced as a deterministic version of simulated annealing in which local search methods are used.

Melechovský, Prins & Calvo (2005) give a mixed integer programming formulation for a problem with non-linear depot costs. The non-linear costs are represented by a piecewise linear function with four intervals. Their heuristic solution approach consists of a combination of a p-median approach to find an initial solution and a hybrid metaheuristic to improve it. The metaheuristic is hybrid in the sense that it merges variable neighborhood search and tabu search principles. The p-median approach starts by randomly choosing starting depots as medians and then assigns customers to these depots based on minimum costs. The solution is improved by first looking for a closed depot to which a full set of customers of an open depot can be assigned at lower cost. After this step, each customer separately is reassigned to a depot if lower costs can be achieved. The p-median method runs till no more improvement is achieved, after which its outcomes form the input for the meta-heuristic approach.

Interpretation

Location-routing models are certainly able to solve the problem of locating hubs and performing vehicle routing in the problem context of Rensa. However, we see no added value of solving the location problem and routing problems in an integrated approach as is done in location-routing. Routing for Rensa is of such variable character, with demand points changing every day, that we do not see the location of facilities influencing the possibility to construct efficient routes.

What remains is the intention of Rensa, to use the crossdocking hubs as locations from which it becomes viable to let trucks drive a second delivery trip. This should increase flexibility and utilization of both truck loading capacity as well as working hours of truck drivers. The class of Multi-trip vehicle routing problems is described in the next section.

3.6 Literature review: Multi-trip vehicle routing problem

When Rensa locates crossdocking hubs, it may become a possibility to let a single truck perform multiple delivery trips. Therefore, this section describes literature on the Vehicle Routing Problem with Multiple trips, or VRPMT.

3.6.1 Definition

The Vehicle Routing Problem with Multiple Trips is an extension of the classical Vehicle Routing Problem (VRP). In the VRPMT each vehicle may perform several routes in the same planning period (Olivera & Viera, 2007). The set of routes performed by a given vehicle constitutes a tour whose total duration cannot exceed a given time limit (Francois, Arda, Crama, & Laporte, 2016). Although many variations of the classical VRP have been studied, such as time windows and heterogeneous vehicles, the aspect of multiple trips has not received much attention in literature.

In many practical applications, the assumption that each vehicle may perform at most one route in the same planning period is unrealistic. When the demands of the customers are large compared to vehicle capacities or when the distances are relatively short, performing more than one route per vehicle may be the only practical solution. In urban areas, where travel times are rather small, it is often the case that after performing short tours vehicles are reloaded and used again i.e. (Fleischmann, 1990), (Francois, Arda, Crama, & Laporte, 2016). Solving the VRPMT not only implies the design of a set of routes, but the assignment of those routes to the available vehicles. This makes the VRPMT a very practical problem, specially at an operational level, in which daily driver schedules must be designed for a fixed vehicle fleet.

3.6.2 Solution methods

The common practice in solving the VRPMT lies in combining VRP and bin packing algorithms (Francois, Arda, Crama, & Laporte, 2016).Vehicle routes are first obtained by applying VRP algorithms. These routes are then assigned to a fleet with a limited number of vehicles, generally by applying bin packing techniques in which each route is viewed as an item whose size corresponds to its duration and each vehicle as a bin of capacity equal to the maximum allowed tour duration. Below the solution techniques used by several research examples that take this VRP and bin packing approach are described.

Fleischmann (1990) solves the VRPMT with a limited heterogenous fleet and with and without time windows. Their solution method is called the Savings procedure for multiple use of vehicles. A savings procedure seems most appropriate for practical problems due to its low computational effort. As the research considers a limited fleet availability, not only the feasibility of single tours, but also the feasibility of the schedule as a whole with respect to the fleet must be considered.

Taillard, Laporte & Gendreau (1996) start their solution approach by generating VRP solution using a tabu search algorithm. Single vehicle routes are stored in a list, and promising routes from this list are used as a starting point for a next tabu search. This process is repeated p times according to an input parameter p. In the next part of the algorithm, a selection of routes from the previous step is used in a search tree. All feasible VRP solutions that can possibly be constructed by combining the selected routes are generated. A set of K feasible VRP solutions is taken to the next step in the algorithm. For each solution in this set, a bin packing problem is solved to obtain a feasible solution to the VRPMT. The heuristic used here is referred to as 'best fit decreasing' in literature.

Olivera & Viera (2007) propose an enhanced tabu search procedure in the form of an adaptive memory algorithm to solve the VRP. The procedure keeps good solutions in a memory so that different components of these good solutions can be used to create new good solutions. Initially, the memory contains components of many different solutions, and local search results will differ substantially among iterations and solutions come from diverse regions of the search space. Later in the procedure, the search is intensified over the promising regions. After the construction of routes using the adaptive memory algorithm, a form of bin packing is used to assign the routes to vehicles.

Salhi & Petch (2004) optimize the maximum driver overtime in their problem. They have an objective function containing regular driving costs and a driver overtime penalty. The solution approach starts by generating VRP solutions using the generalized savings measure defined by Yellow (1970), within the template saving heuristic of Clarke & Wright (1964). For the allocation of routes to vehicles, several bin-packing problems are solved, using a heuristic based on the 'best fit decreasing' algorithm used by Taillard, Laporte & Gendreau (1996). Several improvement modules are used in the algorithm to improve on both driver overtime in the objective function and total solution costs.

Salhi & Petch (2007) describes a Genetic Algorithm for the VRPMT. The model minimizes the maximum overtime for a prescribed number of vehicles. The standard binary chromosome representation of Genetic Algorithms is adapted to fit the VRPMT. Within the steps of the Genetic Algorithm, a savings heuristic and bin packing are used to generate and allocate routes. The algorithm does not outperform benchmark studies, but as it is the first attempt that uses an evolutionary type of method, the researchers feel that the method has strong potential for future development.

Interpretation

Rensa wants to investigate the possibility of driving multiple delivery routes from a hub. The theory on VRPMTs shows that there are numerous suitable approaches available to do this. Most of these approaches use some solution method for the standard VRP, and combine it with a form of bin packing.

The transport planning application of Rensa has some form of savings algorithm in it for the calculation of routes. Research examples of for instance Fleischmann (1990) and Salhi & Petch(2004) also make use of a savings approach. They stress the methods low computational effort, which can also be valuable in the large problem instance we are solving for Rensa. As we strive for an accurate comparison between the current and new logistics networks of Rensa, using a savings algorithm seems like a good fit for this research.

We combine the savings approach with a form of bin packing, as has proven to be well performing in the research examples. We think that these solution techniques are clear, not sensitive to errors and performing stably. This is very important, as our goal is not to achieve transport costs minimization, but rather being able to give an accurate comparison between the current and proposed logistics networks. As explained in the previous paragraph this is another reason for choosing a savings algorithm, as the transport planning application of Rensa has such an algorithm in it too.

3.7 Conclusion

This section derives the most important outcomes from the literature review in sections 3.4, 3.5 and 3.6, with the aim to guide the modeling approach that we will use to analyze the proposed logistics network of Rensa. This modeling approach will be further operationalized and implemented in Chapter 0.

In Subsection 3.4.1, we have seen that the class of discrete location models provides the best fit with our problem context. It serves the goal of staying as close to reality as possible and provides the ability to set specific conditions on facility positions in the form of candidate locations. Among the three branches of discrete location models of Subsection 3.4.2, we have seen that the median-based models clearly provide the best fit with the problem context of this research.

Section 3.5 studied the topic of location-routing models, in which there is dealt with both location and routing aspects in an integrated solution approach (Nagy & Salhi, 2007). We concluded that there is no benefit in solving the location and routing problem in an integrated manner, rather than a sequential approach for our research. The benefit of location-routing problems are mainly in fields where routes have a more fixed character, which is for instance the case in locating retail shops that must be supplied from a given set of locations, or in locating garages for city busses (Maze & Khasnabis, 1985). In these cases the location of facilities directly influences the efficiency that can be achieved in these routes permanently.

From Section 3.6 we concluded that research provides numerous suitable approaches to model the vehicle routing problem with multiple trips. A combination of a savings algorithm with a bin packing approach is chosen for this research. It is a proven technique that has similarities with the transport planning application of Rensa, something that is valuable in comparing the current logistics network with the proposed form.

4. Model definition

This chapter builds further on the chosen solution methods at the end of Chapter 3 and implements them into a complete modeling approach. An answer is given to research question 3:

RQ3 How can the chosen Operations Research modelling techniques be implemented?

The chapter starts with a description of our modeling approach in Section 4.1. The structure of the remainder of this chapter is given at the end of Section 4.1, as it logically follows from our modeling approach.

4.1 Modelling approach

This section describes our modeling approach, derived from the conclusions we drew from literature in Chapter 3. We also elaborate on how we model the logistics network of Rensa and in what sense that differs from reality.

Our modeling approach starts with the formulation of a location model, to choose hub locations for different hub counts and allocate each customer to a hub or Rensa's central DC. In Chapter 3 we concluded that a median-based, discrete location model best fits the problem context of this research. More specifically, we solve a p-median location problem, in which the average distance to customers, weighed with customer demand and transport costs on an edge, is minimized. A p-median model needs candidate locations as input; promising locations where the crossdocking hubs might be placed. We will determine these in the next section.

After the location model, truck routing is considered to give a realistic representation of the transport costs for different hub counts. The current situation will also be considered in the model to form the baseline with which the new networks' costs can be compared. The routing aspect of our problem will be solved using a combination of a savings algorithm and bin packing, with the goal to be able to compare transport costs over different logistics network configurations.

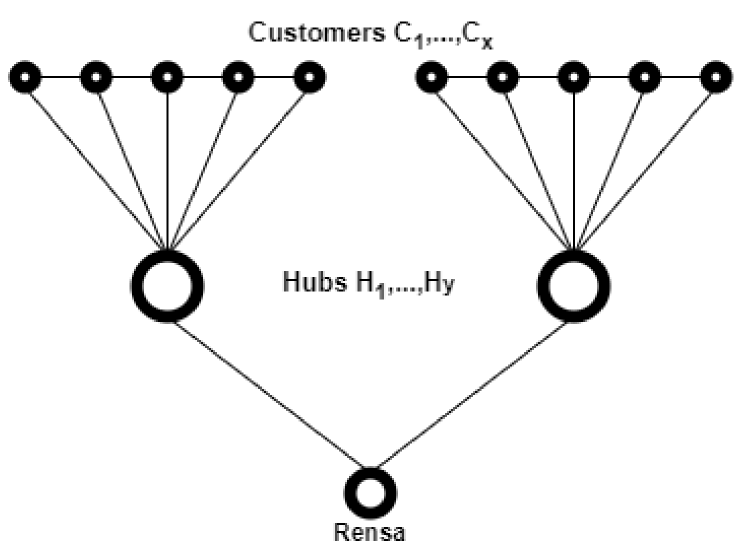


Figure 10 Graph representation of the logistics network used in modelling

There are some differences between Rensa's logistics network in reality, and how it is defined in our modeling approach. We gave a representation of Rensa's logistics network in Figure 6 in Subsection 3.1.2. The graph representation of the adapted logistics network, as used in the location and routing models is represented in Figure 10.

The difference between the network representation in Figure 10 above, and the graph representing the real logistics network of Rensa in Figure 6, is that the warehouses of Rensa in Doetinchem are modelled as one location in our modelling approach. This choice is made for several reasons. Firstly, shuttling to hubs is done in a direct point-to-point manner from individual DCs of Rensa. This means there is only shuttling traffic between the DCs of Rensa for the customers that are still directly delivered from DC4. Further, we are unable to trace which goods came from which specific DC for all customer orders of Rensa, making it difficult to give a proper measure of the transit costs. We are still able to compare our modeling results with the actual transport costs of Rensa. This is because transit within Rensa is seen as warehousing activity, which means that its costs are also stored separately within the warehousing department.

4.2 Location model

This section describes the implementation of the location model as defined in the previous chapter. Subsection 4.2.2 describes how our model differs from the general p-median location problem. Assumptions we made to come to a working solution, are also given in this section. The input data for the model is described in Subsection 4.2.3. In Subsection 4.2.4 the model is defined, followed by a description of the model implementation in Subsection 4.2.5.

4.2.1 Candidate locations

The first important modeling step is to find candidate locations for the new crossdocking facilities. This set of locations forms the input of the location model. Candidate locations are defined in discussion with the head of transport and logistics director of Rensa. The set of locations is chosen such that we are confident that a configuration of hub locations can be chosen among them such that a close to optimal (when every location in the Netherland could be chosen) solution is achieved.

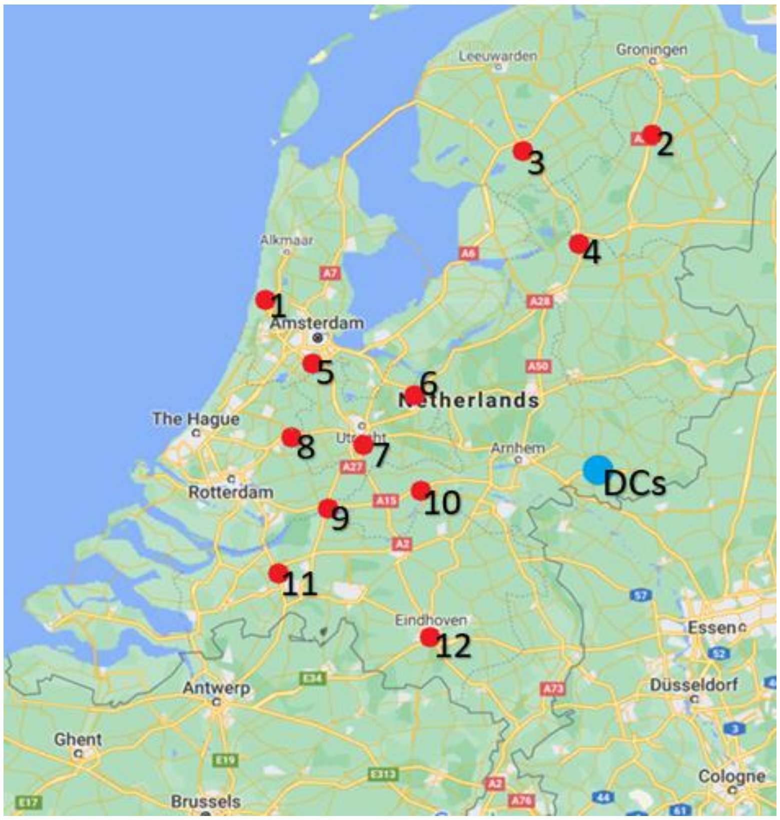
It does not make sense to have crossdocking location very close to the central DCs of Rensa. Customers in this area can be serviced efficiently directly from the central DC in Doetinchem. Locating hubs at the far ends of the country is also not a logical option. The possible service area quickly becomes small or it stretches out back in the direction of the central DCs. The latter would not be logical looking at the shuttling from central DCs to the crossdocking location.

Apart from the objectives already embedded in the costs and distance measures of the location model, there are other factors that influence the location decision for Rensa. An important one is the location of crossdocking hubs relative to existing truck pitches. Keeping existing truck drivers within the organization is important for Rensa, which makes driving times of truck drivers to new hub locations an important measure.

One specific candidate location that is added, is Libra Energy, located in Velsen-Noord. Libra is one of the companies within the Rensa Group. Rensa wants it included in the modelling approach as it may be a possibility in the future to locate a hub at their facility. Its geographic location at the far west of the Netherlands is certainly not optimal from a transport perspective. However, with Libra operating within the Rensa Group, it may still be a promising option.

In practice, the location decision will also heavily depend on practical issues like the availability of a suitable partner or the availability of labor and building land, depending on whether Rensa will outsource the crossdocking activities or not. Having a set of candidate locations gives the possibility to look into the effects of altering a promising hub configuration. This shows what the effect of deviating from an ideal location, due to practical issues, may be.

The chosen candidate locations are visualized on the map of the Netherlands in Figure 11. The location names with the corresponding hub numbers are given next to the map. Locations are numbered from north to south, with the exception that Libra Energy is given hub number 1.



DCs.	Central DCs Doetinchem
1.	Libra Energy Velsen-Noord
2.	Assen
3.	Heerenveen
4.	Meppel
5.	Mijdrecht
6.	Amersfoort
7.	Nieuwegein
8.	Bodegraven
9.	Gorinchem
10.	Tiel
11.	Oosterhout
12.	Best

Figure 11 Candidate locations visualized

4.2.2 Differences and assumptions

This section starts by describing the differences between our model and a standard p-median location problem as defined in the literature studies of Subsection 3.4.2 on discrete location models.

Differences

Our implementation of the p-median location model, as outlined in the remainder of this chapter, differs from the standard p-median model on two aspects. The first difference is that we give the number op open hubs as fixed input to the model, instead of making it a decision variable. The model does not work towards a single optimal solution as output, but calculates the weighted durations of all possible hub configurations for the number of hubs given as input. Doing this, gives us the freedom to further analyze and compare results afterwards, which is helpful as the best hub configuration does not depend on weighed durations only.

Another difference from most model implementations that does not change anything to the methodology itself is that we make use of travel durations instead of distances. This is done because we use a platform named Open Source Routing Machine, or OSRM, which is able to calculate real travel durations of the thousands of demand points we have. More on the OSRM and the architecture behind its model implementation can be found in Section 4.4.

Assumptions

We choose to make the following two assumptions in our modeling approach.

The first assumption is that the seven truck pitches currently located in the Netherlands will close and the new hubs will take over their function. This assumption may be logical when five new hubs are located, but not when we locate only one or two new hubs. Although Rensa may decide to keep certain truck pitches open in the future, their general view is that the hubs take over the functionality of these pitches and that the pitches will be closed in a new situation.

Secondly, the location model does not incur all transport modes that Rensa uses for customer delivery. For the ration between shuttle and customer delivery costs, we assume that shuttling is always done by LHV, where customer delivery is done by box truck. The box truck is Rensa's primary mode of transport. City trailers and delivery vans are used to less extent. In our input data we also make sure that we only use demand data that was transported using box trucks in reality.

4.2.3 Input Data

This section describes the input data of the location model. We can divide the input data in three main branches:

- Customer demand data (measured in volumetric weight)
- Hub and customer coordinates (used as input for the OSRM)
- Transport cost measures of box trucks and LHVs

Customer demand

The first input needed is the ordered demand per customer over 2020. We derive this demand from a database that stores all planned truck stops of Rensa.

The transport planning of Rensa measures the size of customer demands in volumetric weight. Volumetric or dimensional weight differs from the true weight of an order as it accounts for the space that it occupies in transport. This is done, as from a shipping point of view, transporting a large, lightweight package, is more expensive than transporting a small heavy box.

The data with truck stops contains 367,586 records for the year 2020. After cleaning the data (removing empty and zero rows, internal orders, missing addresses and return stops to DC4), we end up with a total of 314,152 truck stops over 2020. These stops belong to 33,484 unique customers. In the location model we thus have 33,484 customer demand points. The volumetric weights off al separate stops are summed per customer to achieve their total volumetric weight ordered over 2020.

Hub and customer coordinates

OSRM, the tool we use to calculate travel distances, needs coordinates as input. This means we need to gather the coordinates of all customer demand points of Rensa.

For the largest share of customer demand points we retrieve the coordinates from transport planning application Smarttour. Still for around 10,000 customers we do not find coordinates there. This is for instance due to delivery addresses that deviate from a customers company address, or construction sites that do not even have an address yet during construction.

The 10,000 missing coordinates are retrieved using Google's Geocoding API (Geocoding API, 2022). This service can be used in Excel and returns coordinates for addresses given as input.

Transport and shuttle costs

Besides weighing durations with customer demands for delivering from hub to customer, our model also incurs the goods flow between central DCs and hubs. This will come back in the model definition in the next subsection, 4.2.4. Shuttling between the DCs and hubs using an LHV is cheaper per unit volumetric weight than customer delivery using box trucks. To take this difference into account, we weigh the shuttling component between DCs and hubs with the ratio of LHV costs compared to box truck costs.

The transport costs of a box truck within Rensa equal €52.85,- per hour. This figure includes direct labor costs, truck costs and overhead. Currently, Rensa does not use LHVs for its logistics, meaning there is no costs figure available within the organization. To get a precise figure we use logistics costs calculations of research institute Panteia (Panteia, 2018), combined with a costs calculation of a logistics partner of Rensa. This results in LHV costs of €78.59,- per hour.

A box truck and LHV have a capacity of 6000 and 18000 units volumetric weight respectively. Combining the costs figures with these capacity leads to a fraction LHV costs compared to box truck costs of 0.5. The goods steam from central DCs to hubs is therefore weighed with a factor 0.5 in the objective function of the location model.

4.2.4 Model definition

This subsection formulates the facility location problem as an integer linear program or ILP. The problem formulation with customers $i \in I$, *i = 1,…,n* and the hubs $h \in H$, *h = 1,…,m* is defined as follows:

$w_i \; = demand\ of\ customer\ node\ i$

$d_{0,h} = travel\ duration\ from\ central\ DC\ to\ hub\ h$

$d_{i,h} = travel\ duration\ from\ customer\ node\ i\ to\ hub\ h$

$C_t \; = transport\ cost\ per\ unit\ duration, per\ unit\ volumetric\ weight\ for\ customer\ delivery$

$C_s = shuttling\ cost\ per\ unit\ duration, per\ unit\ volumetric\ weight\ between\ central\ DC\ and\ hub$

$P \; = number\ of\ open\ hubs\ [1, \dots, 12]$

$$x_h = \begin{cases} 1\ if\ hub\ h\ is\ chosen\ to\ be\ open \\ 0\ if\ not \end{cases}$$

$$y_{i,h} = \begin{cases} 1\ if\ customer\ node\ i\ is\ assigned\ to\ hub\ h \\ 0\ if\ not \end{cases}$$

$$Minimize \qquad \sum_{i \in I} \sum_{h \in H} \left(\left(\frac{C_s}{C_t} \right) w_i d_{1,h} + w_i d_{i,h} \right) y_{i,h} \qquad (1)$$

$$s.t. \quad \sum_{h \in H} x_h = P \quad (2)$$

$$\sum_{h \in H} y_{i,h} = 1 \quad \forall i \in I \quad (3)$$

$$y_{i,h} - x_h \leq 0 \quad \forall i \in I, \forall h \in H \quad (4)$$

$$x_h \in \{0,1\} \quad \forall h \in H \quad (5)$$

$$y_{i,h} \in \{0,1\} \quad \forall i \in I, h \in H \quad (6)$$

The objective function (1) minimizes the sum over all demands and hubs of costs and demand weighted distance under a given hub configuration. The objective function consists of two separate parts for shuttling from DCs to hubs and for distribution from hubs to demand points. The first term $\left(\frac{C_s}{C_t}\right) w_i d_{1,h}$ corresponds to shuttling between DCs and hubs. Besides weighing duration *d* with demand *w*, it has an extra factor $\frac{C_s}{C_t}$ that corrects for the difference in shuttling costs by LHV compared to end distribution by box truck. The second term $w_i h$ defines end distribution and just weighs duration with demand.

The first constraint (2) limits the number of open hubs to the number given as input. Constraint (3) makes sure a customer can only be assigned to one hub. The next constraint (4) states that a demand point can only be assigned to a hub that is open. Finally, constraints (5) and (6) are integrality constraints.

The Python implementation of the location model is further described in Subsection 4.2.

4.2.5 Model description

This subsection further explains the location model, supported by pseudo code of the main functionality. The actual code written in Python can be found in Appendix F: Python code: location model and Appendix G: Python code: routing model.

Durations matrix

The location model starts with the creation of a durations matrix, consisting of durations from each hub to each customer. To do this, requests are made to the OSRM, which is further described in Section 4.4. We also calculate shuttling durations from central DCs to hubs, and update all durations in the durations matrix with corresponding shuttling durations. The shuttling durations are already weighed with the cost difference of transporting one unit of volumetric weight using an LHV compared to a box truck.

Location model

We choose the number of open hubs P as a fixed input. The location model starts with creating a list with all unique hub configurations for the number of open hubs given as input. We then enumerate over all possible combinations of candidate locations that total the number of open hubs. For every combination of open locations, we assign each customer node to the best hub location based on the weighed duration in the objective function.

For every configuration of open hubs, we store the total weighed duration and the allocation of customers to the hub locations. The total weighed duration of a configuration corresponds to the objective function of the ILP defined in Subsection 4.2.4. The pseudo code describing this process is given in Figure 12 below.

$For\ all\ hub\ configurations$

$For\ all\ customers\ i \in I$

$Innitialize\ best\ duration\ as\ distance\ from\ DC4, d_{i,0}$

$For\ all\ hubs\ h \in H$

$If\ duration\ from\ hub\ to\ customer\ d_{i,h} < current\ best\ duration$

$Best\ duration = \ d_{i,h}$

$Store\ best\ allocation\ for\ this\ customer\ and\ update\ total\ configuration\ duration:$

$Configuration\ duration + best\ duration\ d_{i,h} * volumetric\ weight\ w_i$

$$Store\ this\ configurations\ weighed\ duration, \sum_{i\in I}\sum_{h\in H}\left(\left(\frac{c_s}{c_t}\right)w_i d_{1,h} + w_i d_{i,h}\right)y_{i,h}$$

Figure 12 Pseudo code of the location model

The output of the location model consists of a list of all possible hub configurations for a given hub count, along with the total weighed duration for each of them. The best three solutions for each number of opened hubs are used as input to the routing model. They form the promising solution space for which we use the routing model to dive deeper into vehicle routing and the transport costs of a new logistics network.

4.3 Routing model

This section describes the third and main phase of our modelling approach which consists of a model with vehicle routing that has the goal to assess the transport costs in a new logistics network.

Subsection 4.3.1 starts with the assumptions that are made to come to a working solution. The input data for the model is described in Subsection 4.3.2. In Subsection 4.3.3 the model is defined, followed by a description of the model implementation in Subsection 4.3.4.

4.3.1 Assumptions

Several assumptions have been made in the routing model. These assumptions are described in this subsection.

As already explained in Subsection 4.2.2 on assumptions in the location model, we assume that all customer delivery is done using truck type box truck. Goods flow going in Rensa's delivery vans is ignored, because this is such a small goods stream both in terms of volume and costs.

For the routing model we assume that costs on overhead will remain constant over different hub counts. It would not be realistic to assume that overhead costs run linear with total truck duration travelled. Having 10% less driver hours or trucks on the road does not mean that we automatically have 10% less overhead costs. The contrary may be true, as a new and developing network may even ask for more overhead the first years. Because changes are hard to estimate and we do not expect major impact, we assume overhead costs to remain constant.

4.3.2 Input Data

The demand input for the routing model is different from the demand data used in the location model.

In the location model, demand data over a full year is used, summed per customer demand node. The routing model however, needs individual customer orders, resulting in a much bigger data set. The routing model asks for a lot more calculation effort than the location model. The model is run for individual hub configurations and for individual days. Substantial time goes in configuring the model and its input data between configurations and days, as well as in processing the models' results.

In collaboration with Rensa's transport department, we choose to run the routing model based on a five day period of one workweek in November. With data of one week we capture the fluctuations in demand that are present for different weekdays. We choose this specific week as it is representative for demand of Rensa for most weeks of the year. The location decision is made based on average demand and not on exceptional weeks. Network behavior in exceptional weeks is highly dependent on the ability to scale capacity in terms of workforce and trucks, reducing the importance of the factor geographical location of a hub.

As said before, orders that were planned in delivery vans in the original transport planning are not incorporated in our model and therefore deleted from the data set. Further, we make a division between day and night delivery. These are separate goods streams for Rensa currently as well. A small part of day delivery, and complete night delivery are transported by external parties. As Rensa may decide to perform all transport with their own truck fleet in the future, we incorporate these stops performed by external parties in our model as normal stops.

The part of day delivery that is done with Rensa's own trucks, is also stored separately as it is used to validate our model, as explained in Subsection 4.5 on validation.

4.3.3 Model definition

The routing algorithm used in our modeling approach uses a savings algorithm followed by a bin packing procedure, both defined in this subsection.

For the routing part of our modeling approach, we extend the definitions as given in Subsection 4.2.4 with:

$w_i = demand\ of\ customer\ node\ i$

$Q = capacity\ of\ a\ box\ truck$

$d_{i,j} = travel\ duration\ between\ customers\ i, j\ or\ customer\ or\ hub\ i, h$

$s_i = service\ time\ at\ customer\ i$

$s_H = service\ time\ of\ reloading\ at\ a\ hub$

$d_{max} = maximal\ duration\ of\ a\ tour\ (truck\ driver\ working\ time)$

Savings algorithm

We build tours $T = (i_1, \dots, i_K)$, forming a sequence of customer visited by box truck, including the travel duration from hub to customer i_1 and from customer i_K to hub. It is characterized by the quantity shipped $w(T)$ and the duration $d(T)$, including service times. A tour T is feasible if $d(T) \leq d_{max}$ and $w(T) \leq Q$.

A shuttle tour is a tour visiting a single customer. All shuttle tours are supposed feasible, which we have also controlled in the input data of the routing model.

A schedule S is a set of feasible tours $T \in S$, such that every customer is scheduled in exactly one tour. If tours $T_1$and T_2 are in a schedule $T_1, T_2 \in S$, where $T_1 = (i_1, \dots, i_K)$ and $T_2 = (j_1, \dots, j_L)$, the combined tour is given as $T_{1,2} = (i_1, \dots, i_K, j_l, \dots, j_L)$.

The savings algorithm is based on the principle that we can calculate the savings of such a combined tour $T_{1,2}$ over performing the two tours T_1 and T_2separately. If points i and j are at one of the endpoints of their tour, the general savings value, $C(i,j)$, given by linking two points i and j can be given as:

$$C(i,j) = d_{0,i} + d_{0,j} - d_{i,j}$$

Where $d_{0,i}$ is the distance between depot(hub or DC4) and the first point in a route, $d_{0,j}$ is the distance between the last point in a route and depot, and $d_{i,j}$ is the distance between points i and j.

The savings procedure

Below the general outline of the savings procedure for a given hub configuration is defined.

Initial step

For each hub $h \in H$ order all customer pairs (i,j) $(i = 1, \dots, j-1; j = 2, \dots, n)$ on decreasing savings value C(i, j).

Iteration

(a) For the next pair (i,j)proceed to (b) if i and j are the first or last customer in different tours $T_1, T_2 \in S$.

(b) If the combined tour $T^* = T_{1,2}(i,j)$ is feasible, proceed to (c), otherwise to the next iteration.

(c) $S = S \cup \{T^*\} \backslash \{T_1, T_2\}$, the new schedule consists of the old schedule with the new tour T and exclusion of the old separate tours T_1, T_2.

The algorithm iterates until all pairs (i,j) in the savings list have been processed. Any customer not planned in a combined tour is planned in a shuttle tour on its own.

Now we have a complete schedule S we proceed with the bin packing procedure to enable multiple trips.

Bin packing

Below the steps in our bin packing approach are defined.

Two routes T_1, T_2 can be combined if their joint duration fits the maximum duration d_{max}, so $d(T_1) + d(T_2) + s_H \leq d_{max}$. Here s_H denotes the service time of reloading a truck at the hub.

The procedure is based on the best fit of two routes. We measure the fit of two routes by the duration that is left after combining them before we reach d_{max}. We call this measure the fit tightness. For two routes T_1, T_2 the fit tightness is defined as $d(T_1) + d(T_2) + s_H - d_{max}$.

Initial step

Order all tours $T \in S$ on decreasing duration $d(T)$

Iteration

(a) For the next tour T, find tightest fitting feasible combination $T^* = T_{1,2}$, $\min d(T_1) + d(T_2) + s_H - d_{max} \geq 0$. If found continue with (b), otherwise proceed with the next iteration.
(b) $S = S \cup \{T^*\} \backslash \{T_1, T_2\}$, the new schedule consists of the old schedule with the new tour T^* and exclusion of the old separate tours T_1, T_2.

The algorithm stops when we have iterated over all tours T.

We now have a schedule S, that forms the solution of our routing procedure.

4.3.4 Model description

As with the location model, we start with building a durations matrix. For the routing model, shuttling to the hubs does not play a role and is not incurred in the durations matrix. The routing model works with one specific hub configuration as input.

The first step is to create a savings matrix, by looping over all hubs and all possible customer pairs. The savings matrix is sorted in descending order.

$For\ all\ hubs\ h \in H\ in\ the\ given\ configuration$

$For\ customers\ i \in I\ from\ i\ to\ j-1\ allocated\ to\ hub\ h$

$For\ customers\ j \in J\ from\ i+1\ to\ n$

$Calculate\ savings\ \ C(i,j) = d_{0,i} + d_{0,j} - d_{i,j}$

$Sort\ all\ savings\ values\ in\ descending\ order$

Next, we start building the routes. We loop over all customer pairs (i,j) in the savings matrix. There are then three main options for the selected customer pair in the savings matrix:

- Both customers are already in a route
- One of the two customers is already in a route
- Both customers are not yet in a route

If both customers are not in the same route, are on endpoints of a route, and both truck capacity and driver working time are not exceeded, we combine them in a new route. This is the most extensive case as we have to check for both routes if customer i and j are on endpoints of their routes Below is pseudo code on the case of both customers already being in a route.

$For\ all\ customer\ pairs\ i,j\ in\ the\ savings\ matrix$

$If\ both\ customers\ i,j\ are\ already\ in\ a\ route$

$If\ customers\ i,j\ are\ not\ in\ the\ same\ route$

$If\ both\ customers\ i,j\ are\ on\ endpoints\ of\ their\ routes\ T_1,T_2$

$If\ volumetric\ weight\ w_i\ \leq capacity\ Q\ of\ a\ box\ truck$

$and\ if\ the\ combined\ duration\ d_{T_1} + d_{T_2} - C(i,j)\ \leq d_{max}$

Combine routes $T_1, T_2\ of\ points\ i,j\ in\ new\ route\ T^*$

After looping over all entries in the savings matrix, we check if there are unassigned demand points left. These are planned in a tour on their own.

The schedule S that we have obtained from the savings procedure forms the input of the bin packing algorithm. For all planned routes, we loop over all other routes to find the tightest fitting combination of routes that is feasible. If a possible combination is found we combine them, otherwise we try the next route. The pseudo code of the bin packing procedure is given below.

$For\ all\ tours\ \ T \in S$

$For\ all\ other\ tours\ T \in S$

$Initialize\ best\ fit\ tightness$

$If\ the\ combined\ tour\ is\ feasible, (T_1) + d(T_2) + s_H \leq d_{max}$

Calculate fit tightness $d(T_1) + d(T_2) + s_H - d_{max}$

$If\ fit\ tightess < best\ fit\ tightness$

$Best\ fit\ tightness = fit\ tightness$

$If\ a\ fitting\ combination\ is\ found$

$Combine\ tours\ T_1, T_2\ in\ new\ tour\ T^*$

After having looped over all routes, we have our definite schedule S.

4.4 Model architecture

To calculate travel durations between the larger number of customers we have, we use the Open Source Routing Machine or OSRM. This section describes the it architecture behind the location and routing models, built around OSRM. The section gives an answer on research question 3.1:

Rq3.1 How does the architecture behind the implemented routing machine used in our models look like?

To be able to do the number of requests for the large problem instance we have, we cannot rely on the public API of OSRM. Therefore running OSRM locally is needed. We found that there is a lack of comprehensive documentation on how to set up a local OSRM instance working with Python. This subsection provides the architecture behind the local OSRM implementation used in this research.

4.4.1 Open Source Routing Machine (OSRM)

The Open Source Routing Machine or OSRM is able to calculate real travel durations for the thousands of customer locations present in our logistics network.

Besides returning travel duration between two points, OSRM can also perform tasks like finding nearest points, solving a travelling salesman problem and filling a durations matrix, the service we use for the location model. OSRM is built in C++ programming language and relies on geographic data of the OpenStreetMap project to make its calculations.

OSRM is chosen for this research project, as it is very fast and able to work with large numbers of requests (Open Source Routing Machine, sd). Luxen & Vetter (2011) find that OSRM is so fast that calculation effort in routing itself is not a bottleneck anymore, and that other components become obstacles. Limiting components become for instance the retrieval of the route’s geometry and computing driving maneuvers, or bandwidth and network latency.

4.4.2 OSRM architecture

This subsection describes the IT architecture that is set up around the Python implementation of the OSRM. The architecture is presented visually in Figure 13.

Limited use public API

OSRM provides a public API to be used for routing requests. However, this public API is based on fair use, and only intended for testing and demonstration purposes. For problems with only a few locations, resulting in a small number of request, one can make use of this public API. In this case, setting up OSRM can be done rather easily, using a Python wrapper that is available around the OSRM API (Stroetz, 2020).

For the number of requests that are needed for the thousands of customer nodes we have, the public API cannot be used which is why a local OSRM instance is set up in this research.

Running a local OSRM instance

The architecture behind our local OSRM instance is shown schematically in Figure 13.

For setting up a local instance of the OSRM, we found a solution in using Docker. Docker is a tool that provides standardized components, so called containers, to package applications. Docker is a virtual software, meaning it is a separate process that can operate on top of your operating system. Rather

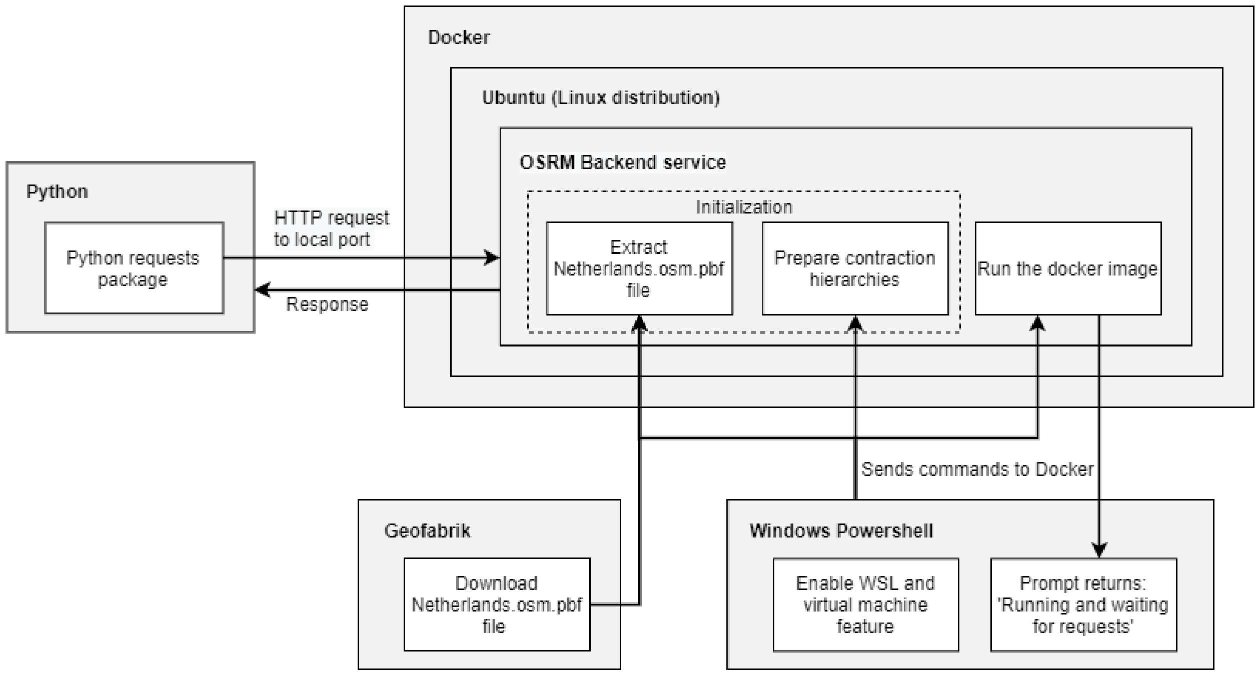


Figure 13 OSRM architecture behind the Python model

than installing software on your local drive, it is installed and ran within the Docker container. This eases sharing and collaboration on applications.

With the OSRM backend Docker image installed, our Docker container contains the functionality we need for our routing requests. As can be seen in Figure 13, Docker really provides a container, in which the functionality is built that is able to handle the routing requests we want to do. When operating, Python can send requests to Docker, and the Docker container contains the functionality to be able to handle these requests.

When Docker is installed and running, the container still needs to go to some initialization steps to be made ready to serve routing requests. The first thing it needs are extracts from a desired geographical region, providing the graph that can be traversed in the routing logic behind OSRM. These extracts are provided by the OpenStreetMap project (GmbH, sd).

The OSRM backend Docker image is built to function on Linux operating system. This means that a Linux distribution should be running in the Docker container. To make this work on a Windows installed device, Windows subsystem for Linux can be used. The specific Linux distribution we run on Windows is Ubuntu. Docker serves as a shell around Ubuntu, meaning we can direct Docker to let Ubuntu execute commands.

The previous paragraphs described all components to let our Docker container function. Having mounted a host-directory in the Docker container so that Docker knows where the files it can use are located, the OSRM backend service can be initialized. This means extracting the map downloaded from OpenStreetMap and preparing the contraction hierarchies, a speed-up technique for the routing algorithm.

The last step is to run the Docker image, which will activate the routing server so that it is ready to use. The specific commands to initialize, run and also use a local OSRM instance can be found in Appendix E: Initializing, running and using Docker images.

4.5 Validation

This subsection elaborates on the validity of the routing model, by comparing our calculations to the real transport costs of Rensa.

Approach

Rensa stores its transport costs broken down into the categories: Rensa trucks, delivery vans, external transport and parcel delivery. Under these categories different cost factors are stored, ranging from labor costs to insurance and damages to trucks. The different costs factors and transport categories add up to the total transport costs per month. To validate our routing model, we compare its results with the actual transport costs of Rensa.

The first validation step we take is to feed the real transport planning of Rensa from the week in November into our model. This way, our distance and cost metrics are used so that we can compare the outcomes with Rensa's actual transport costs. In the second validation step, we take the demand points from the same week in November and let the savings algorithm used in the routing model do the planning and calculations.

In Python, a separate model is built that can calculate the real transport planning of Rensa. We do this based on the same data we use in the routing model later on. Only now, we follow the routes and stop sequence as the trucks of Rensa drove in November 2020. For this calculation, we only use the data on day trucks of Rensa. This category is present among the stored transport costs of Rensa, and we can isolate this category from our data. This makes it possible to do a valid comparison.

Distances are calculated using OSRM, as explained in Subsection 4.4.1. To each first stop in a route, and from each last stop in a route, we add the distance to drive from or back to DC4 in Doetinchem. By doing this, we ignore the fact that some trucks drive back to the truck pitches. There they are combined and shuttle to Doetinchem. Because normally two box truck loads are in one shuttle, we count only half of the distance back to Doetinchem for the routes starting and ending at a truck pitch.

For the routing costs, we use the costs per hour of €52.85 as we did in the location model. These costs were calculated internally at the transport department based on actual cost data and should be very precise.

Outcomes

The actual transport costs of Rensa, compared with the calculation of the real planning and the planning using our savings algorithm, can be found in Table 5. We see that our calculation of the real planning results in costs being 6.6% lower than what was observed in reality by Rensa. Planning with the savings algorithm results in costs being 9.5% lower than reality.

It is difficult to judge if this 6.6% difference is acceptable or not by the numbers alone. However, if we look at the data preparation we have done, the difference can be explained well. We deleted 14.5% of the original demand points of Rensa. Of this 14.5%, 4.5% were stops that registered the return of trucks to DC4 in Doetinchem. This means that we have 10% of deleted demand points left. These can be contributed partly empty rows, volume or volumetric weight being zero or negative and the separate category of point-to-point deliveries. Apart from the point-to-point deliveries that form a very small fraction of these deleted rows, all other deleted rows belong to very small stops or return shipments. That these 10% of deleted demand points together account for around 6.6% of transport costs is very realistic.

The fact that the savings algorithm scores another 3% lower than the calculation of the real planning, shows that we can plan more efficiently than originally done. This is also a logical outcome, as in reality orders come in during the day and planners cannot wait till the moment all orders have arrived to start building routes. This is because the warehouses need the transport planning as soon as possible after the order stop for customers. Also, in reality there may be issues in truck or workforce planning that cause unexpected limitations during the planning process.

Concluding, we are confident that the distance and costs metrics we use in the routing model correspond to reality and are therefore valid. The savings algorithm performs a bit better than actual planning, which can be explained logically and is therefore not a problem.

Table 5 Actual transport costs comparison

	Costs	Difference w.r.t. actual
Actual transport costs	€ 114,670	
Real planning calculated	€ 107,048	-6.6%
Savings algorithm	€ 103,743	-9.5%

4.6 Conclusion

In this chapter, we translated the modeling approach that we derived from literature into a working solution.

Our modeling approach consists of a combination of a location model and routing model. The graph representation of Rensa's logistics network we use in modelling differs from reality in the sense that the warehouses of Rensa in Doetinchem are modelled as one location in our modelling approach, ignoring transit traffic between these DCs. In modeling we make two important assumptions. The first assumption is that the seven truck pitches currently located in the Netherlands will close and the new hubs will take over their function. Secondly, our models do not incur all transport modes that Rensa uses for customer delivery. We assume that shuttling is always done by LHV, where customer delivery is done by box truck.

Location model

The location model is a p-median location model, in which the average distance to customers, weighed with customer demand and transport costs on an edge, is minimized. The model also takes into account shuttling between central DCs and hubs, and makes use of travel durations instead of distances. We use travel durations because we can use a platform named Open Source Routing Machine, or OSRM, which calculates real travel durations between the thousands of demand points we have. The location model works with a set of 12 candidate locations. Input data consists of customer demand data, hub and customer coordinates and transport cost measures of box trucks and LHVs.

The location model is formulated as an ILP and implemented and solved in Python. We use Python for the implementation of our routing model as well. The location model starts with the creation of a durations matrix, using the OSRM. Durations are updated with the shuttling durations from central DCs to hubs. The model then creates a list with all unique hub configurations for the given hub count and assigns each customer to a hub based on the smallest weighed duration. The models' output consists of the total weighed duration of each unique hub configuration.

Routing model

The routing phase of our modeling approach has the goal to give a realistic representation of transport costs for specific hub configurations. The routing problem is solved using a combination of a savings algorithm and bin packing. For the routing model we make the additional assumption that overhead costs will remain constant over different hub counts. Where the location model uses aggregated data, the routing model needs unique customer orders as input, resulting in a much bigger data set. As calculation effort for the routing model is also a lot higher, we do over the period of one week in November.

The savings algorithm is based on the savings that we can calculate of linking two points i, j together instead of having them in separate tours. The savings procedure starts with calculating the savings for all customer pairs, storing them in a savings matrix and ordering them on decreasing savings value. For each customer pair we see if they are already in a tour and if these tours can be combined into a feasible combined tour. We do this until the whole savings list is processed and we have a complete schedule of tours, which forms the input for the bin packing procedure. This procedure looks for separate tours that can be combined into one feasible tour, after which the definite schedule of tours is achieved.

OSRM architecture

To calculate travel durations between the larger number of customers we have, we use the Open Source Routing Machine or OSRM. To be able to do the number of requests for the large problem instance we have, running OSRM locally is needed. As we discovered a lack of comprehensive documentation on setting up a local OSRM instance, we come up with and describe an architecture in this research. We make use of the OSRM backend Docker image, which runs on Linux operating system. With Windows Subsystem for Linux or WSL, we make it run on our Windows computer. Extracts from the OpenStreetMap project are used to provide the graph that can be traversed behind the routing logic of the OSRM. An overview of the OSRM architecture is given in Figure 13.

Validation

To validate the routing model, we do a comparison that is threefold:

- Actual transport costs of the Week in November
- Real transport planning of that week, but using our distance and cost metrics
- Planning done by the routing model using our distance and cost metrics

We find that the calculation of the real transport planning with our metrics results in costs that are 6.6% lower than the actual transport costs. We think that this difference can be contributed to the fact that we had to delete 10% of demand points from the data set that reflected very small stops or return shipments. It looks realistic that these 10% of demand points account for around 6.6% of transport costs.

Looking at our routing model outcomes, we see that transport costs are another 2.9% lower than the transport costs retrieved from calculating the real transport planning with our metrics. We regard it as logical that we are able to plan more efficiently, as we do not have practical issues such as truck or workforce problems that can occur in reality. Altogether, validation gave us the confidence that the routing model is able to give a representative comparison between the current and proposed logistics network design of Rensa.

5. Analysis

This chapter describes the modeling results of both the location model and routing model. Based on our analysis, we aim to answer research question 4:

RQ4 What is the impact of integrating crossdocking hubs in the logistics network on Rensa's outbound logistics?

Section 5.1 describes the outcomes of the location model. The results on transport costs from the routing model are given in Section 5.2. In Section 5.3 sensitivity analysis is performed on different parts of the location and routing models.

5.1 Location model

This section describes the results from running the location model with the 12 candidate locations that we determined in Section4.2.1. We give an answer to research question 4.1:

Rq4.1 What number and location of crossdocking hubs form promising network configurations?

The main goal of the location model is to identify promising hub configurations. This determines which configurations will be further analyzed in the routing model. We elaborate on the best hub configuration for each hub count, according to the location model, in Subsection 5.1.1. In Subsection 5.1.2 we analyze how the goods flow is divided over hubs and DCs for different configurations . The fluctuation of goods flow at individual locations is also discussed in this subsection.

5.1.1 Weighed duration over different hub counts

For each number of hubs and all possible hub configurations, the location model gives the total weighed duration of serving all customer demand nodes as output. We saw this earlier in subsections 4.2.4 and 4.2.5. This subsection elaborates on the best configuration for each hub count. These configurations are listed in Table 6, along with percentual changes in weighed duration when adding hubs. These changes are also shown in the graph of Figure 14.

Both from the percentage change in Table 6 and the graph in Figure 14, we can see that the weighed duration decreases for increasing hub counts. The lowest weighed duration would in theory be reached in the extreme situation of having a hub located at each customer demand point. We can also see that the decline in weighed duration is significant for the first hubs added, but decreases quickly. Where opening a first hub gives a weighed duration decrease of 11.0%, we see an almost linear decline in Figure 14 to a decrease in weighed duration for adding the fourth hub of only 2.2%. At the point of four hubs, the graph flattens until the decrease has almost become zero when adding a twelfth hub location.

How a change in weighed duration ultimately translates to differences in transport costs will show from our routing model. However, the location model shows us the promising region to explore in the routing model. Looking at the previous paragraphs, we decide to at least explore hub counts until the moment the graph in Figure 14 flattens. This would mean that the routing model is run up till 4 or 5 hubs. This way we capture the region where the biggest effects may be expected.

Table 6 Best configuration per hub count

#Hubs	Configuration (Location numbers see table on the right)	Difference (w.r.t previous)	Difference (w.r.t. 0)
0	0		
1	(7)	-11.0%	-11.0%
2	(4, 7)	-8.4%	-18.4%
3	(4, 7, 12)	-5.1%	-22.6%
4	(4, 7, 8, 12)	-2.2%	-24.4%
5	(2, 3, 7, 8, 12)	-2.0%	-25.9%
6	(2, 3, 7, 8, 11, 12)	-1.5%	-27.0%
7	(2, 3, 4, 7, 8, 11, 12)	-1.1%	-27.8%
8	(2, 3, 4, 6, 7, 8, 11, 12)	-1.0%	-28.6%
9	(2, 3, 4, 6, 7, 8, 10, 11, 12)	-0.9%	-29.2%
10	(1, 2, 3, 4, 6, 7, 8, 10, 11, 12)	-0.7%	-29.7%
11	(1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12)	-0.3%	-29.9%
12	(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)	-0.1%	-30.0%

No.	Description
0.	Central DCs Doetinchem
1.	Libra Energy Velsen-Noord
2.	Assen
3.	Heerenveen
4.	Meppel
5.	Mijdrecht
6.	Amersfoort
7.	Nieuwegein
8.	Bodegraven
9.	Gorinchem
10.	Tiel
11.	Oosterhout
12.	Best

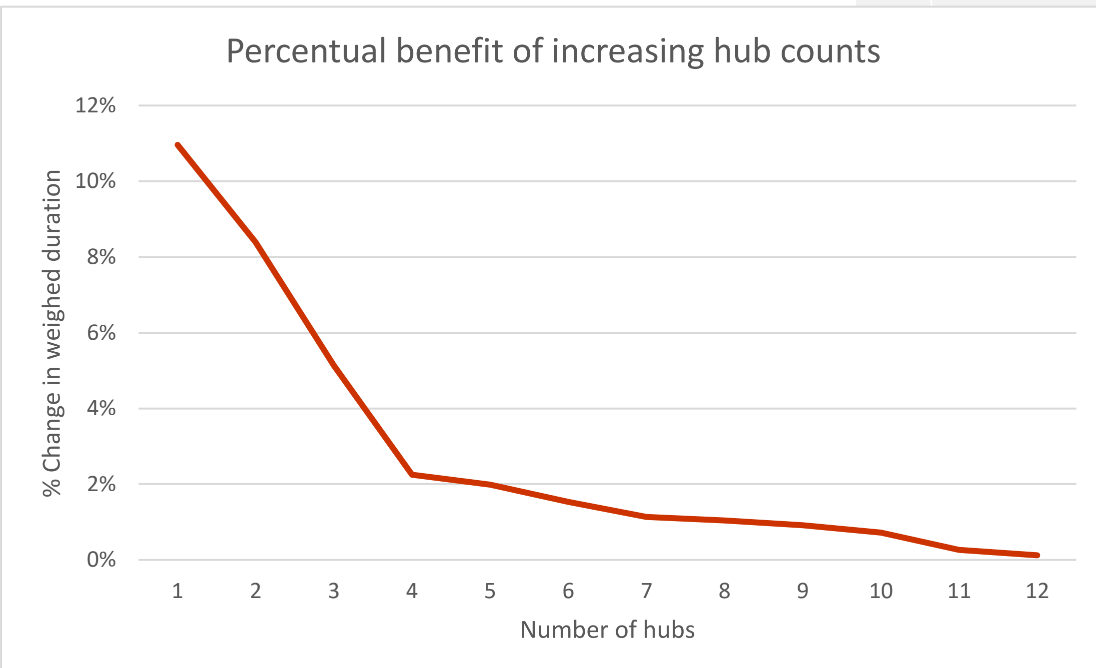


Figure 14 Percentual change of weighed duration for increasing hub counts

5.1.2 Volumetric weight division over hubs

One of the goals of this research is to assess the relief that a hub network may cause at the central DCs of Rensa, especially DC4. We can do this by looking at the sizes of the goods flows via opened locations under different hub configurations. This helps to answer research question 4.2:

Rq4.2 How does the direct goods flow via Rensa's outbound logistics warehouse change under a crossdocking network?

Size of goods flow in our models is measured in terms of volumetric weight, as discussed earlier in Subsection 4.2.3. We can obtain the volumetric weight per hub location by adding up the demands of all customers that were assigned to a certain hub by the location model. The way volumetric weight is divided over the hubs, and the share of goods that is still delivered directly from DC4, gives an indication of the relief that can be achieved at DC4 under different hub configurations. It also gives a view on the size that hubs will have. This becomes relevant when a certain hub size may be needed for the operations on a hub to be feasible.

This volumetric weight division over different locations can be found in Table 7. For each number of hubs, we show the division for the three hub configurations that scored best in terms of weighed duration. Interesting to see is that hub 7. Nieuwegein, handles a larger fraction of total volumetric weight than DC4 in most configurations with two to five hubs. When only DC4 and Nieuwegein (7) are open, DC4 still accounts for the larger share of goods with 57%. Adding Hub 4 (Meppel), we see that the percentage of Nieuwegein does not change. The north of the Netherlands that was served by DC4 under configuration (7) is now served by the hub in Meppel. This results in a percentage of goods flowing from DC4 dropping to 36%, while hub 7 remains at 43%.

Further, we see that the north of the Netherlands is a very stable region in terms of volumetric weight over different hub counts. This already shows from Figure 15, where the light blue area allocated to hub 4 (Meppel) does not change with increasing hub counts. When hubs 2 (Assen) and 3 (Heerenveen) are opened, the customers that were first allocated to Meppel are now divided over these two hubs. The

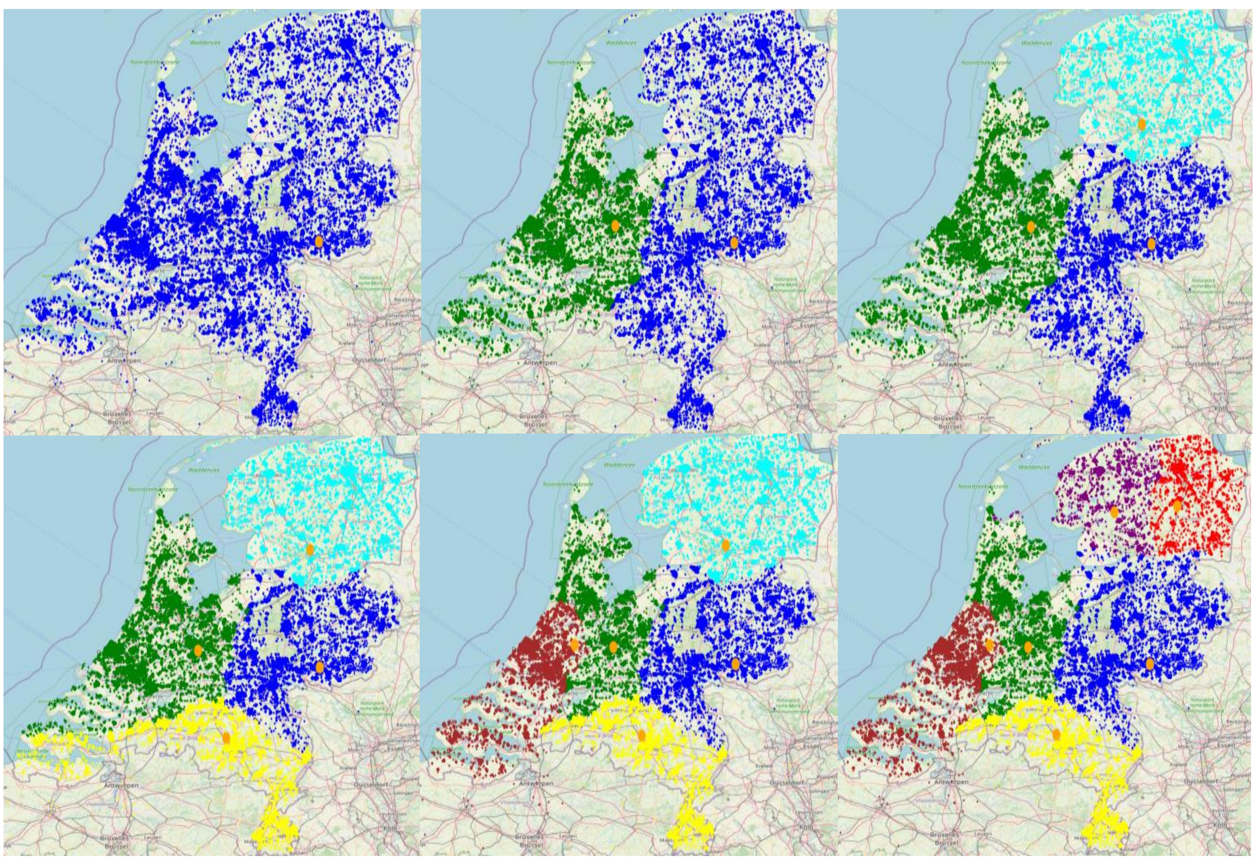


Figure 15 Customer division over hubs visualized for 0, 1, 2, 3, 4, 5 hubs.

Table 7 Customer division over hubs, in percentage of total volumetric weight per hub.

Configuration	DC4	Hub 1	2	3	4	5	6	7	8	9	10	11	12
(7)	57%							43%					
(8)	65%								35%				
(4)	79%				21%								
(4, 7)	36%				21%			43%					
(3, 7)	40%			17%				43%					
(2, 7)	40%		17%					43%					
(4, 7, 12)	23%				21%			36%					20%
(3, 7, 12)	28%			17%				36%					20%
(2, 7, 12)	28%		17%					36%					20%
(4, 7, 8, 12)	23%				21%			21%	17%				17%
(4, 7, 9, 12)	23%				21%			27%		15%			13%
(2, 3, 7, 12)	26%		9%	9%				36%					20%
(2, 3, 7, 8, 12)	26%		9%	9%				21%	17%				17%
(2, 3, 7, 9, 12)	26%		9%	9%				27%		15%			13%
(2, 4, 7, 8, 12)	23%		8%		13%			21%	17%				17%

combined service area roughly stays the same. The same image can be drawn from Table 7. When hub 4 (Meppel) is the only open hub in the north, it always accounts for 21% of total volumetric weight. With only 2 (Assen) or only 3 (Heerenveen) open, this percentage is 17% for each configuration given. With both Assen and Heerenveen open, together they account for 18% of volumetric weight. Geographically these outcomes are logical too. The northern region is cut off on three sides by the North sea and the German border. Only the southern edge of the area is moving up and down a little under different hub configurations.

5.1.3 Volumetric weight fluctuation

By introducing hubs in the logistics network, we are dividing the goods flow of Rensa into smaller streams. This has an effect on fluctuations in demand that are caused at different locations in the network. Bigger demand fluctuations require more capacity in terms of trucks, drivers and warehouse personnel to achieve equal coverage. Also demand fluctuations, depending on their predictability, can make it more difficult to plan capacity and negatively impact truck utilization levels.

We want to know to what extent a more decentral network with hubs suffers more from demand fluctuations than the current network of Rensa. To quantify this we look at the total number of trucks needed to achieve equal coverage at individual locations for different hub counts. Coverage probabilities can be calculated by taking average volumetric weights per location per day and adding one or two standard deviations to them, for respectively 68% and 95% coverage probabilities. When we then translate the found volumetric weight levels to capacity in number of trucks, we can say something about the number of trucks Rensa needs to place at different locations.

The coverage levels roughly match the strategy Rensa has for its truck capacity. We assume that 68% coverage matches the fixed capacity of Rensa's own trucks, while 95% coverage shows the capacity that the flexible layer of third party logistic providers should be able to provide. The remaining probability is

Table 8 Number of trucks needed for 68% and 95% demand coverage probabilities

# Hubs	68% coverage	95% coverage
0	98.9	120.3 (21.4)
1	99.7	121.9 (22.2)
2	100.4	123.3 (22.9)
3	100.9	124.3 (23.4)
4	101.4	125.4 (24.0)
5	101.8	126.2 (24.4)

assumed to be handled as incident, by for instance overloading trucks. Table 8 shows the number of box trucks needed to achieve 68% and 95% coverage levels. The number between the brackets shows the difference between the two levels and corresponds to the size of the flexible layer needed.

We see that a capacity of 98.9 trucks is needed in a situation without hubs, accompanied by a flexible layer of 21.4 trucks. With three hubs open these numbers increase with respectively 2.0% and 9.4% and in a situation with five hubs, fixed and flexible capacity have increased with 3.0% and 13.9% respectively. With five hubs open, there are six more trucks needed in the network to achieve the same coverage level compared to a network without hubs.

5.2 Routing model

This section gives the results of the routing model, that show us what the actual transport costs in different network configurations are. We answer research question 4.3:

Rq4.3 What are the transport costs for various numbers of crossdocking hubs?

Transport costs are calculated based on one week of demand data in November.

5.2.1 Total week costs

This subsection compares the total costs of all hub configurations calculated by the routing model. The total costs over a week of each analyzed hub configuration are listed in Figure 16. The same configurations with percentual differences added can be found in Table 9.

Without hubs, reflecting the current logistics network, the model returns total costs of €116,330- . We see that the biggest marginal costs savings are achieved for adding the first hub. Opening hub 7 (Nieuwegein) results in total transport costs of €110,738-. This is a decrease of 4.0% compared to the no hub situation. Adding one more hub gives minimal week costs of €107,270- . These are achieved when opening hubs 4 (Meppel) and 7 (Nieuwegein), respectively. The opening of a second hub gives us a cost decease of 3.0% over the one hub situation and a cost decrease of 7.0% compared to having no hubs.

Minimum costs with three or four hubs are slightly higher than in a two hub configuration. The lowest week costs are achieved with five hubs, measuring €107,025- . These costs are only 0.22% lower than the costs obtained in a network with two hubs.

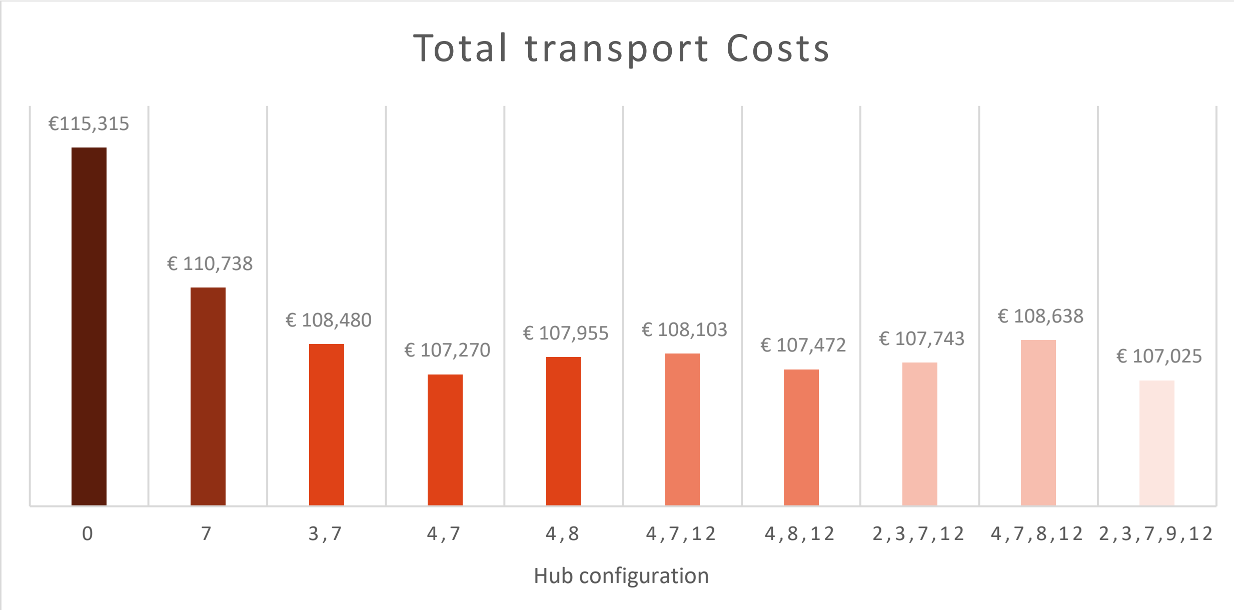


Figure 16 Total transport costs for different hub configurations

Table 9 Total transport costs and percentual differences for calculated configurations

# Hubs	Configuration	Total transport costs		% Difference
0	(0)	€	115,315	
1	(7)	€	110,738	-4.0%
2	(3,7)	€	108,480	-5.9%
2	(4,7)	€	107,270	-7.0%
2	(4,8)	€	107,955	-6.4%
3	(4,7,12)	€	108,103	-6.3%
3	(4,8,12)	€	107,472	-6.8%
4	(2,3,7,12)	€	107,743	-6.6%
4	(4,7,8,12)	€	108,638	-5.8%
5	(2,3,7,9,12)	€	107,025	-7.2%

5.2.2 Cost development per category

Total costs can be divided into expenses for labor, trucks, shuttling and overhead. The cost breakdown for different hub configurations can be found in Figure 17.

Labor cost are the biggest cost category, accounting for roughly half of total transport costs. Labor costs decrease with an increasing number of hubs, representing the ability to plan more efficient tours from the hubs. The development of labor costs follows the same pattern as the total transport costs. We see the biggest cost decrease when adding the first hub, while costs decrease is already much smaller when adding a third hub.

Where labor costs are directly calculated from number of tours that are planned on a given day, truck costs on its turn, are calculated by actual driving time. Actual driving time is given as tour duration

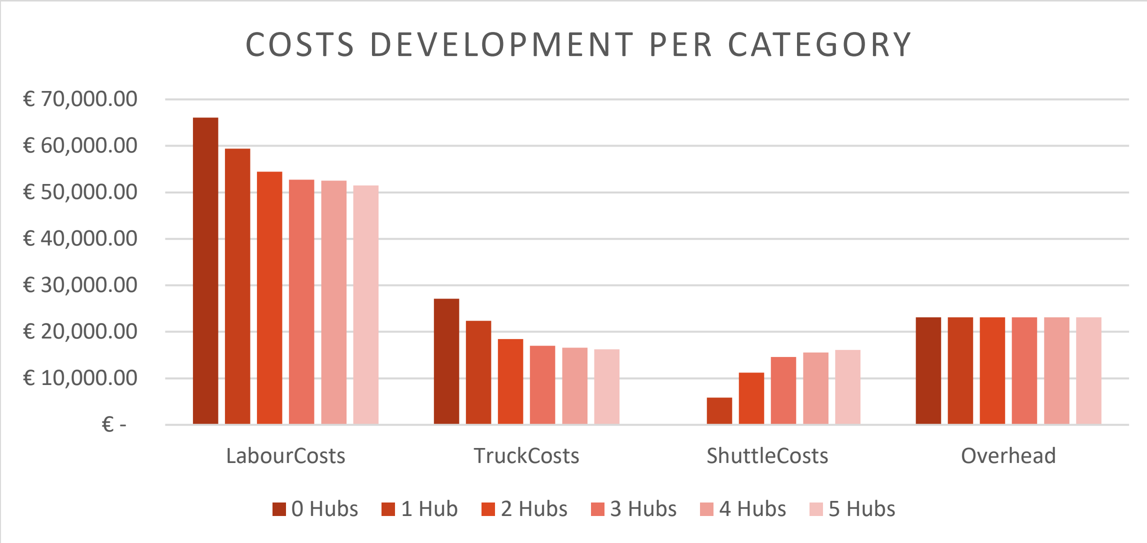


Figure 17 Cost development per category for best hub configuration from one to five hubs

without service or stop time. Labor and truck costs are therefore closely related, which can also be seen from the cost development. Although truck costs only account for a fifth to a quarter of total transport costs, they follow the same pattern as the labor costs do.

Shuttle costs follow the exact opposite pattern from labor and truck costs. Shuttle costs contribution to total transport costs ranges from 7 to 18%. Shuttle costs increase most over the first added hubs. This makes sense, as a lot of demand points change from being assigned to DC4 in Doetinchem to being assigned to a hub. From three hubs on, not many more demand points are withdrawn from DC4.

Further increase in shuttling costs with more than three hubs comes from the fact that shuttling becomes less efficient having more, smaller hubs. With lower volumes per hub, the loading utilization per LHV decreases. However, the cost increase from decreased efficiency is much smaller than the increase for the first few hubs.

5.2.3 Planning efficiency

This subsection elaborates on the number of tours and their efficiency for different hub counts. In Figure 18, the number of tours is given for the best routing model configurations from one to five hubs. The lines in the graph give the total and effective efficiency of these tours. Total efficiency looks at complete tours from depot to customers and back. Effective efficiency ignores driving time between depot and the first and last customer in a tour.

The number of tours decreases for increasing hub counts. This is in line with the fact that the average distance to customers from hubs is much smaller than from DC4 in Doetinchem. The number of tours decreases the most for adding the first and the second hub, from 268 to 241 and 221 respectively. Among the calculated configurations, the least number of tours is achieved having 5 hubs, namely 209. For 3 or 4 hubs, the number of tours respectively numbers 214 and 213, which shows that the decrease in number of tours is small for 3 hubs and more.

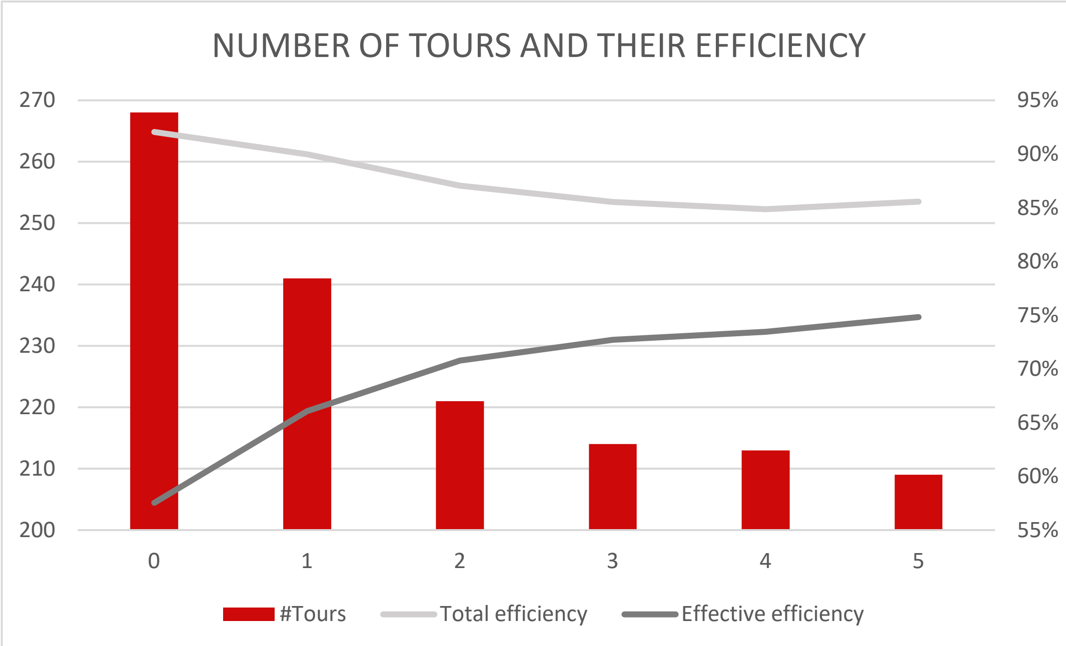


Figure 18 Number of tours and their total and effective efficiency for the best cost configuration for different hub counts

The line of total efficiency in Figure 18 takes the duration of a complete delivery tour and divides this by the 8.75 working hours a trucker dive has available. We see that this measure starts at 92% with no hubs and decreases tot 84% or 85% with three to five hubs.

Where the total efficiency is decreasing when opening hubs, it is interesting to look at the efficiency of tours without driving times between depots and the first and last customers. The average distance to customers becomes smaller by opening hubs, which means that these driving times to first and from last customer also decrease. Looking at the effective efficiency in Figure 18, we see that it measures 58% without hubs, and increases to 75% when having 5 hubs open.

5.3 Sensitivity analysis

A logistics network using crossdocking hubs depends on the ability to transport the large goods streams between central DC and crossdocking hubs in a cost efficient manner. In this research, this shuttling is done using LHVs. This subsection investigates the impact of varying LHV costs and loading utilization on the total transport costs.

The location model in this research uses transport costs per hour, divided by volumetric weight capacity as input. The ratio between LHV costs and box truck costs is then calculated as follows:

$$Costratio = (LHV costsper\ hour / LHV\ capacity)/(boxtruckcostsper\ hour / box\ truckcapacity)$$

For the modelling results given in Section 5.1, the location model uses maximum capacity of both LHV and box truck as input. The formula then looks as follows:

$$(78.74/18000)/(53.75/6000) = 0.488$$

This ratio shows that in our standard modelling approach, the LHV costs per hour per unit volumetric weight are broadly half the cost of the same parameter for a box truck.

For sensitivity analysis we vary the LHV capacity and LHV shuttle costs conform the scenarios given in Table 10. The corresponding cost ratios are calculated using a box truck capacity of 85% instead of 100%, corresponding to utilizations we achieved in our routing model. We first use these scenarios as input for the location model, resulting in a customer allocation. Afterwards, the routing model is run with the location model outputs. The shuttling costs from the routing model are calculated using the LHV capacity corresponding to the LHV utilization from the corresponding scenario.

All calculations done in this subsection use hub configuration (4,7,12), with hubs open in Meppel, Nieuwegein and Best. This configuration is used as it came out as one of the most promising configurations in modelling results.

To visualize the differences in location model output, Figure 20 and Figure 19 show the customer allocation for scenarios 1 and 6 with the lowest and highest cost ratios. We see that as LHV transport becomes more expensive compared to a box truck, more customers are directly served from central DC Doetinchem.

Results from running the different scenarios in our routing model can be found in *Figure 21*. The data used is from the same week in November as used in our standard modeling approach. The figure shows the total week costs of each scenario and the percentual savings over the no hub scenario, i.e. the current logistics network of Rensa. Also, a base scenario is added, corresponding to our normal modeling approach. There, a LHV utilization of 88% was achieved.

Table 10 Sensitivity analysis: LHV costs and utilization scenarios

Scenario	Shuttle costs	LHV utilization	Cost ratio
1.	-10%	100%	0.37
2.	Normal	100%	0.42
3.	Normal	90%	0.46
4.	Normal	80%	0.52
5.	Normal	70%	0.59
6.	+10%	70%	0.65

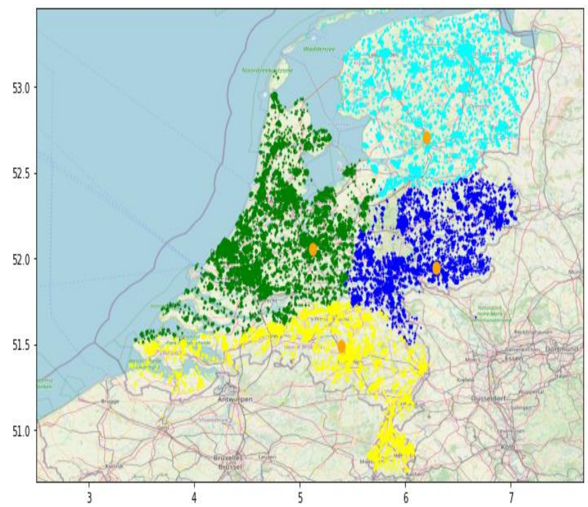


Figure 20 Customer allocation; Scenario 1

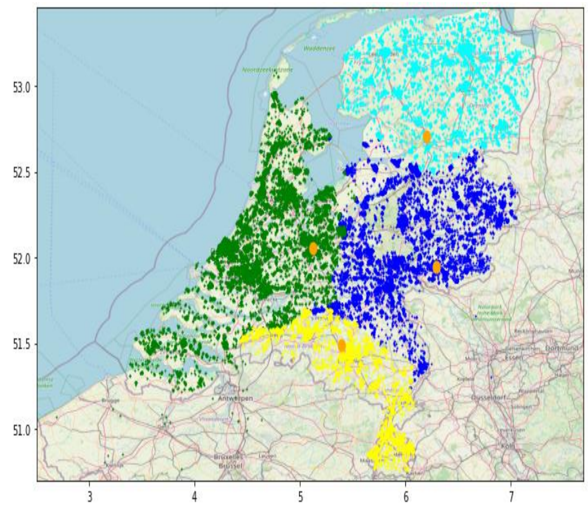


Figure 19 Customer allocation; Scenario 6

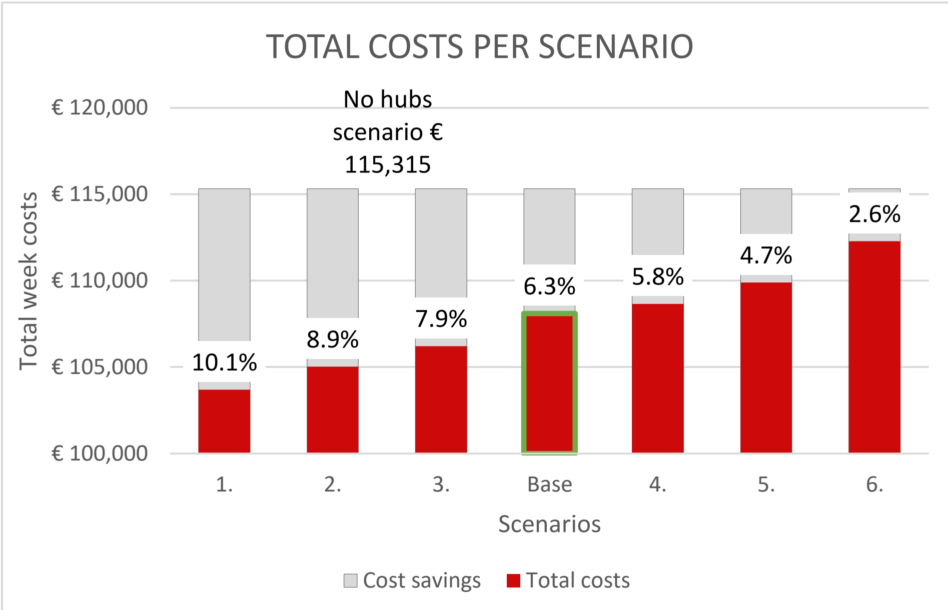


Figure 21 Sensitivity analysis: total costs per scenario

The results show that when we are able to achieve a 100% utilization of our LHV's, transport cost savings over a no hub situation can increase to 8.9 or even 10.1%, when also the LHV costs turn out cheaper. On the other side of the spectrum, cost savings decrease to 4.7% when we achieve a LHV utilization of only 70%. When LHV costs turn out more expensive also, these cost savings become 2.6%.

Concluding, we see that achieving a high LHV loading utilization has a big impact on potential cost savings. This is in line with our expectations. However, we see that even at a LHV utilization rate of only 70%, a hub network can still be more cost efficient than Rensa's current logistics network in terms of transport costs.

5.4 Conclusion

This section concludes this chapter on modeling results. Conclusions are organized along the sections of this chapter on results of the location model, routing model and sensitivity analysis.

Location model

The results on the location model in Section 5.1, show us that a significant decline in weighed duration, which measures travel duration weighed with demand and transport costs, can be achieved by opening hubs. However, the volumetric weight decline quickly decreases when more hubs are added. The first hub gives a weighed duration decrease of 11.0% where adding the fourth only results in another 2.2% decrease. From the location model results we conclude that running the routing model for up to four or five hubs is enough the capture the best network configurations.

Subsection 5.1.2 looks at the division of goods flow over individual locations for different hub configurations and shows that direct goods flow via DC4 can decrease by 43 to 74% in the best hub configurations up to five hubs. It is interesting to see that in most configurations hub 7 (Nieuwegein) serves a larger share of goods than DC4 does.

The northern region of the Netherlands is really stable in terms of demand that is served by hub locations there. Under different configurations the fraction of total goods demand served by this region only ranges from 17 to 21%. From a geographical point of view this is quite logical as the northern region is bounded on three sides by the North sea and the German border.

Looking at the fluctuations of volumetric weight at individual locations, we see that in a network with three hubs, fixed and flexible truck capacity need to be 2.0% and 9.4% higher respectively to achieve equal coverage probability compared to a network without hubs. In in network with five hubs, these numbers increase to 3.0% and 13.9% respectively. With five hubs open, there are six more trucks needed in the network on a total of 126 trucks to achieve equal coverage levels.

Routing model

Section 5.2 looks at the transport costs for different hub configurations as calculated by the routing model. Transport decrease most for opening the first and second hub, with 4.0 and 7.0% respectively compared a network without hubs. By opening more than two hubs, transport costs do not decrease significantly anymore. Transport costs reach their minimum having 5 hubs open, but these costs are only 0.22% lower than the costs obtained having two hubs.

Labor cost are the biggest cost category, accounting for roughly half of total transport costs. Labor costs and truck costs decrease with an increasing number of hubs, representing the ability to plan more efficient tours from the hubs. Shuttling costs follow the exact opposite pattern, increasing when hubs are opened, where the increase is the biggest for the first opened hub.

Subsection 5.2.3 shows us that increasing the number of hubs results in less planned delivery tours. The number of tours decrease from 260 to 209 in a network with five hubs. The cause of the decreasing number of tours does not show in the total efficiency of potential truck driver time used. This number decreases from 92% to 84% when opening five hubs. However, when we discard the driving time from and to the depot, we see where the increased efficiency is achieved. The effective efficiency as we call it increases from 58 to 75%.

Sensitivity analysis

The sensitivity analysis in Section 5.3 looks into the effect of varying shuttling costs and LZV loading utilization on total transport costs. The scenario’s plotted vary LHV costs with -10 and +10%, and LHV utilization from 70 till 100%.

LHV costs and utilization should be an important focus in possible further steps in investigating or implementing hubs. There is a difference of 8.3% between our best (-10% cost, 100% utilization) and worst (+10% costs, 70% utilization) tested scenario. This is a big difference than can influence the viability of the whole hub network. Comforting is the fact that even in the worst tested scenario, transport costs are still 2.6% lower compared to the current network.

From this, we can conclude that we are able to use the driving time of truck drivers more efficiently by opening hubs. The decreased average distance to customer plays an important role here, as driving times to first and from last customer to depots become much smaller. The fact that the total efficiency decreases when opening hubs, may come from the fact that tour planning becomes less efficient when the pool of customers to be planned becomes smaller. This is something that Rensa should keep an eye on in further steps taken on implementing hubs.

6. Conclusions and recommendations

This chapter concludes this research on a logistics network involving crossdocking hubs at Rensa. Section 6.1 gives the most important conclusions. Limitations on our research approach are discussed in Section 6.2. Section 6.3 states the recommendations for Rensa that we take from this research, and Section 6.4 sketches opportunities for further research on the topic.

6.1 Conclusions

This section gives the most important conclusions of this research. Subsections 6.1.1 starts with a recap on our problem context and research approach. Subsections 6.1.2, 6.1.3 and 6.1.4 give the most important conclusions of this research.

6.1.1 Recap

This research was triggered by growth of Rensa and the pressure that this growth puts on its logistical processes. There is pressure on physical storage space, but especially on outbound logistics capacity. Problems that Rensa faces here are limited floor space, insufficient loading docks and strict truck departure times that put pressure on the loading process and leave no space for mistakes.

Earlier research on Rensa's logistics network performed by Groters (2020) finds that improvement potential within the current logistics infrastructure is limited and forms only a short term solution for the growing organization. Also, emission rules for city centers are becoming stricter and Rensa has started a transition towards an electrified truck fleet. The current logistics network, centralized around Doetinchem, is not a good fit for electrical (city center) distribution. Groters recommends investigating a different logistics network design for Rensa. The network he proposes locates a number of warehouses performing crossdocking throughout the Netherlands.

The first part of this research looks into challenges and opportunities that come with having a crossdocking network. Literature review combined with interviews and a survey give us a broader understanding of the problem context. Literature review on location models, location-routing models and the multi-trip vehicle routing problem leads to a modelling approach. The approach consists of a combination of a location model and a routing model, that are solved sequentially. We use the location model to find promising hub configurations and assess these in our routing model. This leads to transport costs and division and variation of volumetric weights over different locations. We expand our analysis with sensitivity analysis on shuttle costs and LHV utilization.

From our modeling results together with the research on crossdocking networks performed earlier we can draw a number of conclusions that will be outlined in the next subsections.

6.1.2 Transport costs decrease when opening crossdocking hubs

The implementation of crossdocking hubs in the logistics network of Rensa results in lower transport costs. With the opening of two hubs, transport costs savings of 7.0% can be achieved.

Transport costs decrease significantly by adding a first and second hub, but marginal savings decrease sharply after that. To be able to divide the Netherlands in logical, equal zones, two hubs is the minimal number to open. Opening more than two hubs only leads to minor extra transport costs savings. Opening more than two hubs should then be an consideration that is supported by other factors than transport cost savings. This may be in the average distance to customers and city centers, in relation to electrical distribution, or in growth expectations of Rensa. Another important reason may be the relief of DC4, which is discussed in the next subsection.

From Subsection 5.3, we stress the importance of efficient organization of the shuttling between central DCs and hubs. Shuttling costs that turn out 10% higher or lower, combined with LHV loading utilizations that vary between 70 and 100%, result in cost savings that can increase to 10.1% or decrease to 2.6%. The fact that transport costs savings are still 2.6% compared to the current network, under the worst tested scenario, shows that the efficiency of shuttling forms no decisive risk for viability of the whole network.

6.1.3 Opening hubs causes relief at DC4 as outbound logistics warehouse

DC4 is relieved significantly by the implementation of crossdocking hubs in the logistics network of Rensa. The fraction of goods still distributed directly via DC4 ranges from 57% in a one hub situation, to only 23% with three or more hubs in the network. We are able to reduce the goods flow directly distributed from DC4 with three quarters.

A hub location in Nieuwegein, located centrally in the Netherlands, handles a bigger fraction of goods in most of the hub configurations than DC4 does. This is an eye opener for Rensa on the impact that a new logistics network will have, but also on the relief that can be caused at DC4 as outbound logistics warehouse.

6.1.4 Trucks can be used more flexibly and efficiently

The factor that came out as most important in our context analysis was more flexible and efficient use of trucks from hubs. Decreasing transport costs, less planned delivery tours and their increased efficiency show us that we are indeed able to operate trucks from a hub more flexibly and efficiently.

The total number of tours driven decrease from 268 with no hubs, to 221 and 209 with two or five hubs opened respectively. With opening hubs, a growing fraction of tours has an extra reloading stop at the hub, which helps to maximize driver time utilization. We found that the driver time utilization increases from 58% with no hubs, to 71 or 75% with two or five hubs opened respectively. We see that more flexible operation of both trucks and truck drivers is possible from hub locations.

6.2 Research limitations

This section discusses limitations that may be of influence on the results of this research project.

6.2.1 Focus on transport costs

This research focuses on transport costs and research factors directly related to transport. We did this to focus our research efforts in the most relevant direction costs wise, keeping in mind the requirement of relieving DC4.

The shift of workload from DC4 to elsewhere was an essential part of a new logistics network design for Rensa. This research showed that a network using crossdocking hubs can serve this purpose, while even making transport more efficient. Due to our focus on transport costs solely, the question that actually remains for Rensa, is if it is possible to operate a hub location for the transport costs savings that we have calculated in this research. However, this still does not covers the real dilemma Rensa faces. Relieving DC4 can be of such vital importance, that also when in totality not cost-feasible, there may still be chosen for crossdocking hubs in the logistics network.

For Rensa, to be able to answer the question if the give transport savings in this research are sufficient to operate a hub network, it needs to gain better understanding of the factors beyond transport. Setting up a successful organization on a hub, designing warehousing processes and investigating impacts on the commercial departments of Rensa are examples of relevant topics that can influence the success of a crossdocking network. In further steps towards a new logistics network, such topics need further investigation and have to be in scope.

6.2.2 Modelling period

In a trade off between strength of our modelling results on the one side and calculation effort, dataset size and time to run our models on the other, we chose to take a modeling period of one week for our routing model. The chosen week of data is representative for biggest part of the year for Rensa.

We support making the location decision based on demand data represent for most of the year rather than peak demands. Network performance in extremes is highly dependent on the ability to scale the operation and capacity to match demand, reducing the impact of the location decision.

Where our main focus was on the location decision, we do not want to neglect the importance of the logistics network to deal with peak demands. Our research results show that more truck capacity is needed in a network with crossdocking hubs to maintain similar service levels. Designing an operation that is flexibly able to cope with peak demands should be a field of focus for Rensa when proceeding with the implementation of crossdocking hubs in their logistics network.

6.2.3 Number of candidate locations

Based on similar trade offs as the modeling period, we choose to use a set op 12 candidate location for our location and routing models. Choosing a bigger set of candidate location could have brought us closer to the true optimum configuration. This would have resulted in higher costs savings for a new logistics network than we have achieved now, only making the case for a new network design stronger.

Having more candidate locations, and modeling more hub configurations would have given Rensa the opportunity to evaluate more specific locations and configurations based on changing practical reasons. However, with the models now in place, redoing calculations based on specific desires is always a possibility in the future.

6.2.4 Future growth not taken into account

We ran our models with fixed data sets and did not take future growth of Rensa into account.

In a growing organization, the pressure to relief DC4 as outbound warehouse only becomes stronger. As there are no possibilities for expansion of the current facility, a successful operation comes in danger. Also the requirements of successfully operating a network with crossdocking hubs; organizing efficient shuttling between central DCs and hubs and setting up a stable organization that is able to deal with peak demands, become stronger when the total demand pool grows.

Where the case for a hub network only becomes stronger with growing demand, the optimal configuration of hubs will change. The minimal number of hubs of two that we advise stays in place. However, with higher demand biggest transport cost savings may be achieved with a higher number of hubs. Proceeding with a hub network, Rensa should consider the ability to add hub locations in the future, or at least be able to serve increased demand with existing locations.

6.3 Recommendations

This section gives several recommendations to Rensa based on the outcomes of this research project. Recommendations are mainly focused on the further steps to be taken from the point where we stand now.

6.3.1 Start an implementation project in an interdepartmental team

This research showed that by implementing crossdocking hubs, we are able to relief DC4 as outbound warehouse while making transport more efficient. The pressure on outbound capacity at DC4 will only become more severe with an organization that is growing. We advise Rensa to take next steps towards implementation of crossdocking hubs in the logistics network.

The main stakeholder in the implementation of a new logistics network and driver of this research on crossdocking hubs is the transport department of Rensa. Up till this point other departments are not involved in the process. This research showed that success of a new logistics network is dependent on many factors beyond the scope on transport. Without support of other logistical departments such as warehousing and logistics support, but also commercial departments, implementing and operating a successful network is impossible. We advise Rensa to take next steps on new logistics network in an interdepartmental team.

6.3.2 Start small and do a phased implementation

When Rensa decides to proceed with the implementation of a logistics networking using crossdocking hubs, we advise to do a phased implementation and start with one hub location.

Starting with a single hub location gives the opportunity to experience all operational changes and impacts in practice, while at the same time keeping risks and investments relatively low and maintaining the opportunity to choose another direction in the end. We think that fully investigating and quantifying all operational impacts upfront is practically impossible.

As each hub serves a dedicated service area, experiencing the success of a network in terms of transport costs can be done with a single hub opened. With the first hub operational, and the relief of DC4 that causes, the priority of opening next hubs can be determined.

6.4 Future research

The need for further research is already stressed in the previous section. Here, we give several concrete examples of further research directions.

6.4.1 Broader research on operational aspects

As already pointed out in Subsection 6.2.1, this research focused on transport costs in a new logistics network. Where this is a decisive factor on the viability of the network, other aspects play an important role as well. Several aspects more related to warehousing and operational design that can be topics of future research are given here.

A first topic that can be investigated is the impact of a hub network on the warehousing processing in central DCs of Rensa. Processes here will change under a new logistics network. This is especially the case for DC4, from which only a quarter or third of goods are directly distributed in a new situation. LHVs are loading throughout the day and need to depart to hubs on time. This shifts workload and impacts deadlines on order picking. Staff planning needs to be adapted to this new situation.

A second topic of interest lies in efficient shuttling from central DCs to hubs. Starting point is that shuttling is done point-to-point from each of Rensa's central DCs to hubs. However, to achieve high loading utilization rates it may be necessary to be flexible in where and when LHVs depart. Also, the capacity of the LHV itself may need to be flexible.

Where the processes on exiting warehouses change, a completely new organization and new processes need to be set up at the hubs. Having a stable organization at a hub was already raised as concern during context analysis in this research. The design of an organization at the hub requires further research, among with aspects as the size of an organization and late working times on a hub in relation to stability and vulnerability of the operation as a whole.

6.4.2 Differentiate truck loading times more

Currently, trucks for day and night distribution use a dedicated expedition area where all trucks are loaded at the same time and depart together. This means that all ordered goods that Rensa ships during a day occupy the expedition area in front of the docks at the same time. Multiple use of dock doors is difficult, as that automatically results in a longer loading process in total and pressure on timely departure of trucks.

A possibility to make multiple use of expedition space and dock doors possible, and in that way relief DC4, that we find worth investigating lies in more differentiated departure times of trucks. With the transport planning being sent to the WMS completely around 18.30PM this is difficult now. The possibility could be investigated to finalize some delivery tours already during the day and sending them to the WMS, to make it possible to load these trucks earlier. Changing the actual departure time of the truck is not even necessary, but the floor space and dock door can be used again for loading other trucks afterwards.

In an extreme situation Rensa could plan and perform delivery tours the whole day. Regardless the time of the day, transport planning could finalize any delivery tour that meets some efficiency benchmark. Pressure on expedition space and dock doors can be relieved drastically this way, but it also means a totally different logistics process that may come with various obstacles.

Bibliograph

Aamer, A. (2018). An application of lean assessment in a cross-docking distribution center. *Proceedings of the International Conference on Industrial Engineering and Operations Management* (pp. 976-984). Bandung: IEOM Society.

Aboulaid, H., Jardini, B., Sedqui, A., Elkayl, M., Britel, M., Amri, M., & Lyahyaoui, A. (2016). Process re-engineering and success of integration projects of information technologies case study: Process modeling of a cross docking platform of a car manufacturer. *Proceedings of the 3rd IEEE International Conference on Logistics Operations Management, GOL 2016.* Fes: IEEE.

Ahangamage, N., Niwunhella, D., Vidanagamachchi, K., & Wickramarachchi, R. (2020). Implementing a cross-docking system in a warehouse-A systematic review of literature. *Proceedings of the International Conference on Industrial Engineering and Operations Management* (pp. 1005-1012). Dubai: IEOM Society.

Averbakh, I., & Berman, O. (2002). Minmax p-Traveling Salesmen Location Problems on a Tree. *Annals of Operations Research*, 55-68.

Barreto, S., Ferreira, C., José, P., & Sousa Santos, B. (2007). Using clustering analysis in a capacitatedlocation-routing problem. *European Journal of Operational Research* , 968-977.

Barsing, P., Daultani, Y., Vaidya, O., & Kumar, S. (2018). Cross-docking Centre Location in a Supply Chain Network: A Social Network Analysis Approach. *Global Business Review*, 218-234.

Bednar, L., & Strohmeier, E. (1979). Lagerstandortoptimierung und Fuhrparkeinsatzplanung in der KonsumgiiterJndustrie. *Zeitschrift für Operations Research*, 89-104.

Benbitour, M., Sahin, E., & Barbieri, A. (2016). A comparison of different cross-docking organizations in a JIT manufacturing system; Application in the automotive industry. *Proceedings of 2015 International Conference on Industrial Engineering and Systems Management* (pp. 1390-1396). Seville: IEEE.

Benrqya, Y. (2019). Costs and benefits of using cross-docking in the retail supply chain: A case study of an FMCG company. *International Journal of Retail and Distribution Management*, 412-432.

Bilinska-Reformat, K., & Dewalska-Opitek, A. (2013). Telematics of Delivery Optimization Based on Cross Docking Concept in Poland. *Communications in Computer and Information Science* (pp. 200-207). Katowice-Ustron: Springer Verlag.

Braglia, M., Castellano, D., Gallo, M., & Romagnoli, G. (2019). A visual planning solution to streamline the processes of hybrid cross-dockings. *Production Planning and Control*, 33-47.

Branco, I., & Coelho, J. (1990). The Hamiltonian p-median problem. *European Journal of Operational Research*, 86-95.

Bruns, A. (1998). *Zweistufige Standordplanung unter Berücksichtigung von Tourenplanungsaspekten - Primale Heuristiken und Lokale Suchverfahren.* Sankt Gallen University.

Buakum, D., & Wisittipanich, W. (2019). A literature review and further research direction in cross-docking. *Proceedings of the International Conference on Industrial Engineering and Operations Management* (pp. 471-481). Chiang Mai: IEOM Society.

Buijs, P., Szirbik, N., Meyer, G., & Wortmann, J. (2012). Situation awareness for improved operational control in cross docks: An illustrative case study. *IFAC Proceedings Volumes* (pp. 1196-1201). Bucharest: IFAC.

Buijs, P., Vis, I., & Carlo, H. (2014). Synchronization in cross-docking networks: A research classification and framework. *European Journal of Operational Research*, 593-608.

Cavalla, D., Flack, J., & Jennings, R. (1997). Modern Strategy for Preclinical Pharmaceutical R&D.

Chopra, S. (2003). Designing the distribution network in a supply chain. *Transportation Research Part E: Logistics and Transportation Review*, 123-140.

Chopra, S., & Meindl, P. (2016). *Supply Chain Management.* Pearson.

Clarke, G., & Wright, J. (1964). Scheduling of vehicles for a central depot to a number of delivery points. *Operations research*, 568-581.

Current, J., Daskin, M., & Schilling, D. (2001). 3 Discrete Network Location Models. In Z. Drezner, & H. Hamacher, *Facility Location: Applications and Theory.* Springer.

Daganzo, C. (2005). *Logistics Systems Analysis.* Springer.

Daskin, M. S. (2008). What You Should Know About Location Modeling. *Naval Research Logistics*, 283-294.

de Carvalho, P., de Araújo Kalid, R., Rodriguez, J., & Santiago, S. (2019). Interactions among stakeholders in the processes of city logistics: a systematic review of the literature. *Scientometrics*, 567-607.

Drezner, Z. (1982). Fast algorithms for the round trip location problem. *IEE Transactions*, 243-248.

Fleischmann, B. (1990). The vehicle routing problem with multiple use of the vehicles.

Francois, V., Arda, Y., Crama, Y., & Laporte, G. (2016). Large neighborhood search for multi-trip vehicle routing. *European Journal of Operational Research*, 422-441.

Galbreth, M., Hill, J., & Handley, S. (2008). AN INVESTIGATION OF THE VALUE OF CROSS-DOCKING FOR SUPPLY CHAIN MANAGEMENT. *Journal of Business Logistics*, 225-239.

Geocoding API. (2022, 02 11). Retrieved from Google Developers: https://developers.google.com/maps/documentation/geocoding

GmbH, G. (n.d.). *OpenStreetMap Data Extracts*. Retrieved from geofabrik.de: http://download.geofabrik.de/

Goetschalckx, M. (n.d.). Chapter 7. Continuous Point Location. In *Logistics Systems Design.* Atlanta: Georgia Institute of Technology.

Groters, R. (2020). *Optimizing the outbound logistics network.* Doetinchem.

Hansen, P., Hegedahl, B., Hjortkjaer, S., & Obel, B. (1994). A heuristic solution to the warehouse location-routing problem. *European Journal of Operational Research*, 111-127.

Heerkens, H., & van Winden, A. (2012). *Geen probleem.* Van Winden Communicatie.

Khan, H., Hussainy, S., Khan, K., Khan, E., Sharif, M., & Tariq, S. (2017). The implications of cross-docking in the manufacturing sector of Pakistan. *International Journal of Supply Chain Management*, 131-136.

Kinnear, E. (1997). Is there any magic in cross-docking? *Supply Chain Management*, 49-52.

Konrad, S., & Boysen, N. (2011). Cross-docking. *Journal of Management Control*, 129-137.

Kreng, V., & Chen, F. (2008). The benefits of a cross-docking delivery strategy: A supply chain collaboration approach. *Production Planning and Control*, 229-241.

Ladier, A., & Alpan, G. (2016). Cross-docking operations: Current research versus industry practice. *Omega*, 145-162.

Lalwani, C., Disney, S., & Naim, M. (2006). On assessing the sensitivity to uncertainty in distribution network design. *International Journal of Physical Distribution & Logistics Management*, 5-21.

Laporte, G., & Nobert, Y. (1981). An exact algorithm for minimizing routing and operating costs in depot location. *European Journal of Operational Research*, 224-226.

Laporte, G., Nobert, Y., & Arpin, D. (1986). An exact algorithm for solving a capacitated location-routing problem. *Annals of Operations Research*, 293-310.

Laporte, G., Nobert, Y., & Pelletier, P. (1983). Hamiltonian location problems . *European Journal of Operational Research*, 82-89.

Lin, C., Chow, C., & Chen, A. (2002). A location-routing-loading problem for bill delivery services. *Computers & Industrial Engineering*, 5-25.

Logistics valley. (2021). *Project Zero Emissie vrachtwagen 2021-2022*. Retrieved from Logistics valley: https://www.logisticsvalley.nl/nl/projecten/project-zero-emissie-vrachtwagen-2021-2022

Luxen, D., & Vetter, C. (2011). Real-Time Routing with OpenStreetMap data. *H.3.5 Information Systems Applications, G.2.2 Mathematics of Computing*, 513-516.

Maloni, M., & Carter, C. (2006). Opportunities for research in third-party logistics. *Transportation Journal*, 23-38.

Maze, T., & Khasnabis, S. (1985). Bus garage location planning with dynamic vehicle assignments: A methodology. *Transportation Research Part B: Methodological*, 1-13.

Melechovský, J., Prins, C., & Wolfler Calvo, R. (2005). A Metaheuristic to Solve a Location-Routing. *Journal of Heuristics*, 375-391.

Melo, M., Nickel, S., & Saldanha-da-Gama, F. (2009). Facility loation and supply chain management - A review. *European Journal of Operational Research*, 401-412.

Nagy, G., & Salhi, S. (1996). Nested Heuristic Methods for the Location-Routeing problem. *Journal of the Operational Research Society*, 1166-1174.

Nagy, G., & Salhi, S. (2007). Location-routing: Issues, models and methods. *European Journal of Operational Research*, 649-672.

Naunthong, P., & Sud-On, P. (2020). Understanding Organizational Characteristics for Cross-Docking Adoption: A Case Study of Thai Industries. *IEEE 7th International Conference on Industrial Engineering and Applications* (pp. 864-867). Bangkok: IEEE.

Olivera, A., & Viera, O. (2007). Adaptive memory programming for the vehicle routing problemwith multiple trips. *cumputers & operations research*, 28-47.

Open Source Routing Machine. (n.d.). Retrieved from Openstreemap Wiki: https://bit.ly/3qvYs1M

Owen, H. S., & Daskin, S. M. (1998). Strategic facility location: A review. *European Journal of Operational Research*, 423-447.

Paciarotti, C., Ciarapica, F., & Giacchetta, G. (2009). Cross-docking project: A case study. *Renewable Energy and Power Quality Journal*, 563-568.

Panteia. (2018). *Kostencalculaties in het beroepsgoederenvervoer over de weg.* Zoetermeer: Panteia.

Perl, J., & Daskin, M. S. (1985). Warehouse location-routing problem. *Transportation Research*, 381-396.

Piachaud, B. (2002). Outsourcing in the pharmaceutical manufacturing process: an examination of the CRO experience. *Technovation*, 81-90.

Prodhon, C., & Prins, C. (2014). A survey of recent research on loaction-routing problems. *European Journal of Operational Research*, 1-17.

QGIS. (n.d.). Retrieved from QGIS.org: https://www.qgis.org/en/site/index.html

Reeves Jr., K. (2007). Supply chain governance: A case of cross dock management in the automotive industry. *IEEE Transactions on Engineering Management*, 455-467.

Rensa. (2019). *Family breidt uit met nieuw distributiecentrum in Doetinchem*. Retrieved from Rensa: https://www.rensa.nl/nieuws/rensa-family-nieuw-distributiecentrum-doetinchem

Rensa Family Company. (n.d.).

ReVelle, C., & Eiselt, H. (2005). Location analysis: A synthesis and survey. *European Journal of Operational Research* , 1-19.

Robuste, F., Estrada, M., & Lopez-Pita, A. (2004). Formulas for Estimating Average Distance Traveled in Vehicle Routing Problems in Elliptic Zones. *Transportation Research Record*.

Rodrigue, J. (2013). The Geography of Transport Systems.

Salhi, S., & Fraser, M. (1996). An integrated heuristic approach for the combined location vehicle fleet mix problem. *Studies in Locational Analysis*, 3-21.

Salhi, S., & Nagy, G. (2009). Local improvement in planar facility location using vehicle routing. *Annals of Operations Research*, 287-296.

Schaffer, B. (1998). Cross docking can increase efficiency. *Automatic I.D. News*.

Srivastava, R. (1993). Alternate solution procedures for the location-routing problem. *Omega*, 497-506.

Srivastava, R., & Benton, W. (1990). THE LOCATION-ROUTING PROBLEM: CONSIDERATIONS IN PHYSICAL DISTRIBUTION SYSTEM DESIGN. *Computers Opns Res.*, 427-435.

Stephan, K., & Boysen, N. (2011). Cross-docking. *Journal of Management Control*, 129-137.

Stroetz, U. (2020, 10 6). *ustroetz/python-osrm*. Retrieved from github.com: https://github.com/ustroetz/python-osrm

Taillard, É. D., Laporte, G., & Gendreau, M. (1996). Vehicle Routeing with Multiple Use of Vehicles. *Journal of the Operational Research Society*, 1065-1070.

Transport & Logistiek. (2021, March 1). *Rensa Family Company houdt pilot met elektrische vrachtwagen*. Retrieved from Transport & logistiek: https://transportlogistiek.nl/elektrisch/rensa-family-company-houdt-pilot-met-elektrische-vrachtwagen/

Tuzun, D., & Burke, L. I. (1999). A two-phase tabu search approach to the location routing problem. *European Journal of Operational Research*, 87-89.

Van Belle, J., Valckenaers, P., & Cattrysse, D. (2012). Cross-docking: State of the art. *Omega*, 827-846.

Vis, I., & Roodbergen, K. (2011). Layout and control policies for cross docking operations. *Computers and Industrial Engineering*, 911-919.

Vogt, J. (2010). THE SUCCESSFUL CROSS-DOCK BASED SUPPLY CHAIN. *Journal of Business Logistics*, 99-119.

Vogt, J., & Pienaar, W. (2010). Implementation of cross-docks. *Corporate Ownership and Control*, 474-484.

Walha, F., Chaabane, S., Bekrar, A., & Loukil, T. (2014). The cross docking under uncertainty: State of the art. *2014 International Conference on Advanced Logistics and Transport* (pp. 330-335). Hammamet: IEEE.

Wanke, P., & Zinn, W. (2004). Strategic logistics decision making. *International Journal of Physical Distribution & Logistics Management*, 466-478.

Wu, T., Low, C., & Bai, J. (2002). Heuristic solutions to multi-depot location-routing problems. *Computers and Operations Research*, 1393-1415.

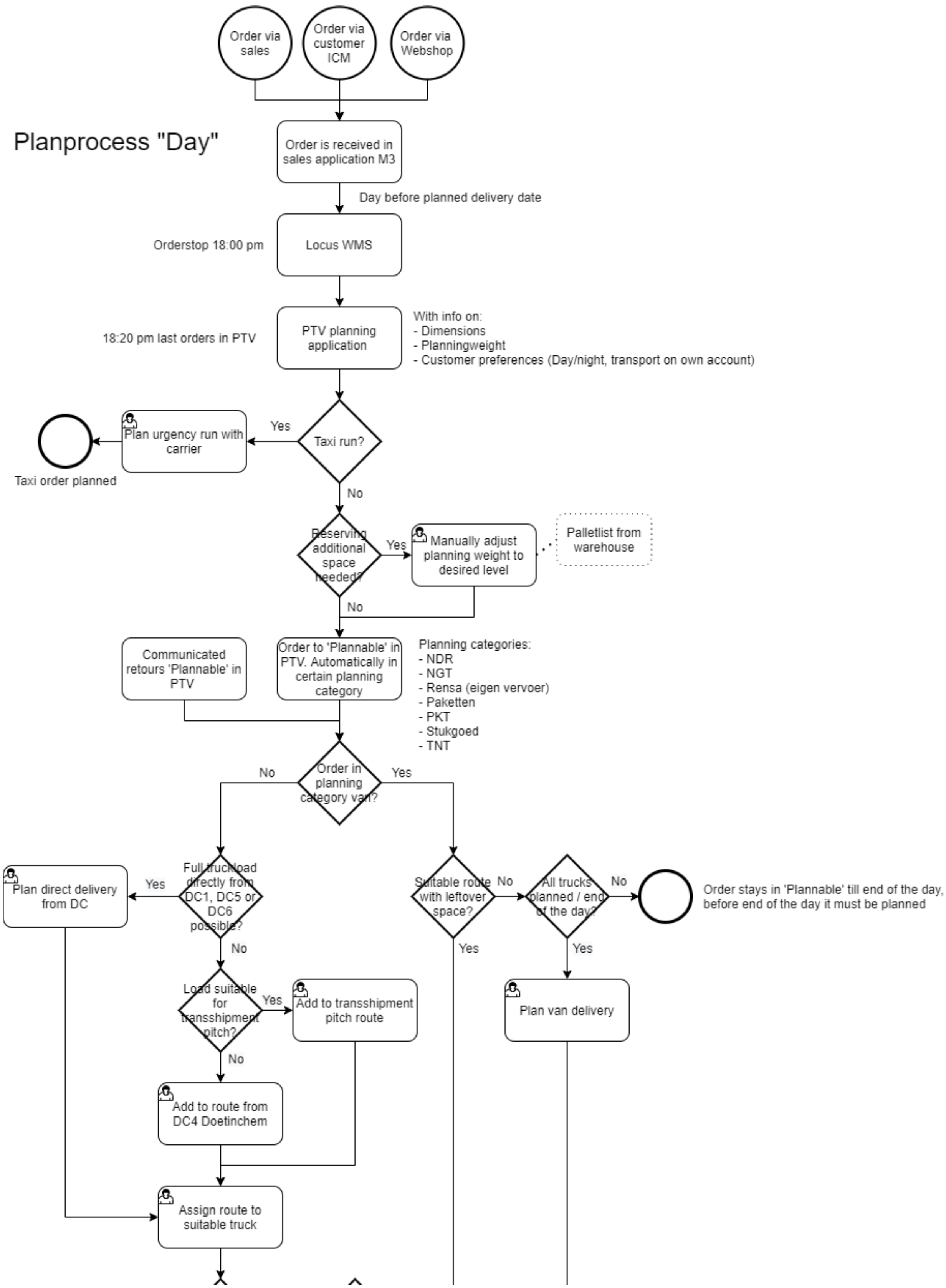
Yellow, P. (1970). A Computational Modification to the Savings Method of Vehicle Scheduling. *Journal of operations research society*, 281-283.

Appendix

Appendix A: Truck pitches

Figure 22 Truck pitches

Appendix B: Transport planning



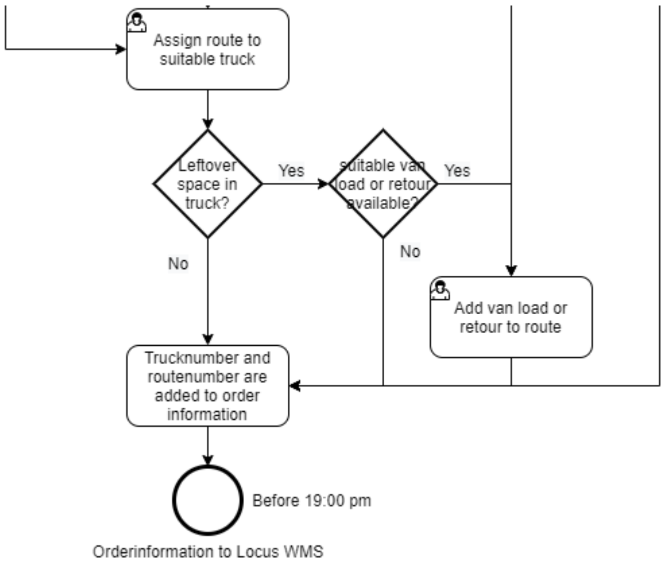


Figure 23 Transport planning process 'Day'

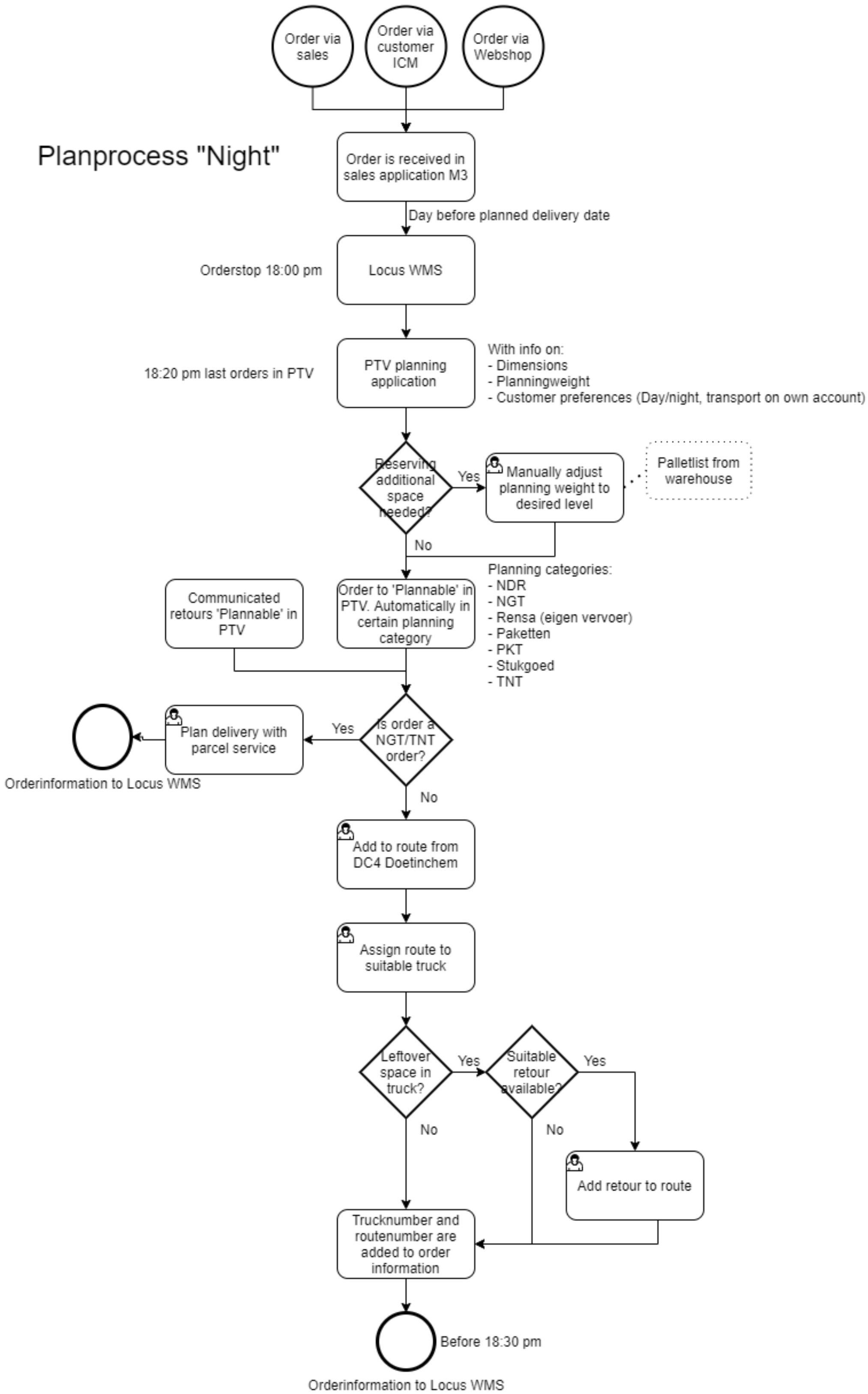


Figure 24 Transport planning process 'Night'

Appendix C: Literature review

Table 11 Literature sources

#	Source	Title	Year
1.	(Naunthong & Sud-On, 2020)	Understanding Organizational Characteristics for Cross-Docking Adoption: A Case Study of Thai Industries	2020
2.	(Ahangamage, Niwunhella, Vidanagamachchi, & Wickramarachchi, 2020)	Implementing a cross-docking system in a warehouse-A systematic review of literature	2020
3.	(Benrqya, 2019)	Costs and benefits of using cross-docking in the retail supply chain: A case study of an FMCG company	2019
4.	(Braglia, Castellano, Gallo, & Romagnoli, 2019)	A visual planning solution to streamline the processes of hybrid cross-dockings	2019
5.	(Buakum & Wisittipanich, 2019)	A literature review and further research direction in cross-docking	2019
6.	(Barsing, Daultani, Vaidya, & Kumar, 2018)	Cross-docking Centre Location in a Supply Chain Network: A Social Network Analysis Approach	2018
7.	(Aamer, 2018)	An application of lean assessment in a cross-docking distribution center	2018
8.	(Khan, et al., 2017)	The implications of cross-docking in the manufacturing sector of Pakistan	2017
9.	(Aboulaid, et al., 2016)	Process re-engineering and success of integration projects of information technologies case study: Process modeling of a cross docking platform of a car manufacturer	2016
10.	(Ladier & Alpan, 2016)	Cross-docking operations: Current research versus industry practice	2016
11.	(Buijs, Vis, & Carlo, Synchronization in cross-docking networks: A research classification and framework, 2014)	Synchronization in cross-docking networks: A research classification and framework	2014
12.	(Walha, Chaabane, Bekrar, & Loukil, 2014)	The cross docking under uncertainty: State of the art	2014
13.	(Bilinska-Reformat & Dewalska-Opitek, 2013)	Telematics of Delivery Optimization Based on Cross Docking Concept in Poland	2013
14.	(Van Belle, Valckenaers, & Cattrysse, 2012)	Cross-docking: State of the art	2012
15.	(Buijs, Szirbik, Meyer, & Wortmann, 2012)	Situation awareness for improved operational control in cross docks: An illustrative case study	2012
16.	(Vis & Roodbergen, 2011)	Layout and control policies for cross docking operations	2011
17.	(Stephan & Boysen, 2011)	Cross-docking	2011
18.	(Vogt, THE SUCCESSFUL CROSS-DOCK BASED SUPPLY	THE SUCCESSFUL CROSS-DOCK BASED SUPPLY CHAIN	2010

	CHAIN, 2010)		
19.	(Vogt & Pienaar, Implementation of cross-docks, 2010)	Implementation of cross-docks	2010
20.	(Paciarotti, Ciarapica, & Giacchetta, 2009)	Cross-docking project: A case study	2009
21.	(Kreng & Chen, 2008)	The benefits of a cross-docking delivery strategy: A supply chain collaboration approach	2008
22.	(Galbreth, Hill, & Handley, 2008)	AN INVESTIGATION OF THE VALUE OF CROSS-DOCKING FOR SUPPLY CHAIN MANAGEMENT	2008
23.	(Reeves Jr., 2007)	Supply chain governance: A case of cross dock management in the automotive industry	2007
24.	(Schaffer, 1998)	Cross docking can increase efficiency	1998
25.	(Kinnear, 1997)	Is there any magic in cross-docking?	1997

Table 12 Contents per literature source

#	Research	Opportunities/Benefits	Threats/Challenges	Relevance
1.	Caste study: survey among 300 Thai firms on organizational characteristics regarding crossdocking	- Goods can be delivered in smaller volumes, with a faster speed and more frequently	- Preferred in a medium to high, constant demand-rate - distribution operation requires high investment costs on both physical infrastructure and technologies to support fast distribution operation - a high number of vehicles is needed to be always ready for frequent and large shipment deliveries - inbound and outbound transportation is needed to be highly synchronized	Survey across many different industries, company sizes and network characteristics.
2.	Literature review: identify processes involved, benefits, complexities and factors to consider in implementation	- mixing or re-sorting out at a close distribution center which will enhance the driver effectiveness - reduces inventory costs, storage space and handling costs - accelerates cash flow and minimizes cycle times due to the elimination of a storage point in the supply chain - more frequent deliveries possible - increase the handling capacity of the warehouse - efficient consolidation of products - improved resource utilization	- Suitable for predictable, high demand and high cubic volume flow products - Imbalance between incoming and outgoing load - Higher stock out probability - Cross-dock terminal will be more labor intensive since cargo must be sorted and moved quickly to relevant destinations - Operation of cross-docking needs adequate material carriers within the warehouse - effective information flow is a necessity	General applicable
3.	Case study: costs/benefits of implementing the crossdocking strategy in a retail supply chain	- inventory costs, storage space and handling costs - accelerates cash flow and minimizes cycle times due to the elimination of a storage point in the supply chain - Reduced time to market	- reduction in the quantities ordered combined with an increase in the frequency of deliveries greatly increased the suppliers handling cost - as the quantities to be delivered are increasingly reduced, the transportation costs increase	Focus on retail supply chain does not match our situation. There is no delivery after retail DC

			- literature shows also that other products with a stable demand, longer shelf life and low value are more suitable for a cross-docking strategy with more focus on cost minimization - double picking activities in the supply chain	
4.	Investigate the implementation of lean and visual planning techniques from the specific context of an Italian oil and gas company	- both reduce inventory and improve customer satisfaction	- crossdocking works best for companies that either serve many stores or distribute large volumes of products - different types of wastes such as missed synchronizations, temporary storage, unevenness in workforce allocation and quality issues	Most part of the research is about visual planning and not relevant in general
5.	Literature review: operation problems and metaheuristic methods to solve those problems	- cutting inventory costs while increasing the goods flow and shortening the shipping cycle - order picking and storage activity could be minimized or eliminated - reducing transportation cost - increasing customer satisfaction	- requires local and network-wide operations to be synchronized - appropriate software should be implemented simultaneously - Task scheduling between inbound and outbound dock doors of cross-docking is a complex scheduling problem in itself since multiple resources need to be coordinated and each worker needs to be scheduled in detail	Mainly focused on task scheduling in the cross dock
6.	Social network analysis model description, use case	- reduce the turnaround time and decrease the time-related risks in the supply chain - reducing inventory costs, increasing inventory turns, consolidating transportation, increasing throughput and reducing operation costs associated with eliminating unnecessary handling and storage - increase inventory velocity - enhanced customer service levels and improvement in the supplier relationships	- requires the supplier to respond reliably and quickly within a short lead time - The schedule depends on right information of the arrival and the departure times, and the destinations where product need to be delivered - Short lead time needs to be ensured by knowing the correct information about arrival and departure of goods that are going to be delivered - Transportation process should be well coordinated to minimize any uncertainty	About food supply chain, but challenges are general applicable
7.	Case study: Crossdocking DC operations assessed using lean principles		- synchronization within the cross-docking operation, to meet the necessary service levels - Set-up Time in DC operations, as represented in the shipping dock doors set-up and inbound receiving lanes set-up, could potentially increase overheads, decrease machine utilization, decrease labor productivity, and increase queue time - Waste was manifested in the long queues, poor staff planning, lack of floor supervision and direction, overproduction, shift and breaks transitions, and in the lack of discipline and sense of urgency - Excessive Move Time, such as stocking depalletization, could potentially increase material handling costs and labor - automation can lead to a tremendous	More focused on lean than on the crossdocking principle

			amount of waste, and a decrease in service levels	
8.	Case study: role of crossdocking in the FMCG sector in Pakistan	- Reduce inventory, reduce lead times, reduction of costs and consolidation of supplies - consolidation of goods - improves customer order time through transferring goods from receiving towards the shipment vehicle directly without storing it in the warehouse - improvement of customer services	- trust, cooperation and proper communication among the supply chain partners - effectiveness varies industry-to-industry and organization-to-organization - controlling the risk factors associated with not holding inventory, some safety stock or hybrid system is advised - Reduction of labor costs depends on the nature of labor. Permanently vs outsourcing - Increased chance of damage (although the contrary also has been pointed out in earlier research)	Specifically in Pakistan, but challenges can be applicable in general, also parts based on literature
9.	Case study: reengineering of a platform of cross docking		- The efficiency of the delivery of the goods is obtained by the coordination of the flow of information and of the physical flows - the sharing of information and, therefore, the visibility of process is a condition for the success of the continued replenishment and cross-docking	Focus is on process reengineering and IT. Crossdocking is just the case study
10	Framework that helps comparing literature review with on-field observations	- cutting inventory costs while increasing the goods flow and shortening the shipping cycle	- congestion in the facility - Truck processing time deviation: trucks are forced to arrive earlier or leave later than planned, because it is not possible to start their unloading or finish their loading on time - Delayed arrivals - The uncertainty of the activity volumes, the diversity of skills required (licenses to drive the different material handling equipment, training on the Warehouse Management System, etc.), the large range of operating hours, and the frequent use of temporary workers make scheduling a difficult task - it is necessary to have an overview of the workload to plan the upcoming days	
11	Framework specifying the interdependencies between different crossdocking problem aspects to support future research. Two real life illustrative problems	- cross-docking can realize transport efficiencies at reduced material handling and storage costs by eliminating the storage and order picking activities from the main warehouse operations - enable the consolidation of multiple less-than-truckload shipments to realize economies in transportation costs - enhance distribution responsiveness	- local operations at the cross-dock are tightly coupled with distribution activities elsewhere in the supply chain due to the absence of a storage buffer - the design and coordination of cross-docking operations requires a holistic approach, which aims to synchronize local and network-wide operations - Lateral interdependencies are particularly important in the design and coordination of cross-docking operations due to the absence of a storage buffer inside a cross-dock - total distance traveled and the congestions that appear on-route between dock doors are important factors regarding the operational performance of a cross-dock - The paper asserts that the absence of a storage buffer inside a cross-dock translates into tightly coupled local and network-wide cross-docking operations	Broad classification of the whole crossdocking concepts and decision problems field
12	Literature review:		- External uncertainties: the arrival times of	Relevant as

	crossdocking under uncertainty		trucks do not follow a precise schedule, the number of inbound trailers can increase or decrease according to the suppliers, the type and quantity of goods vary enormously along the day, the month and the year - Internal uncertainties: the departure times of the trucks can be restricted, the processing time of containers have a major impact on the cross dock operations, the resources such as forklifts handling materials and conveyors may be available or not, number of available outbound trucks can disturb planning	uncertainty is not described explicitly a lot
13	Case study on crossdocking in retail chain in Poland	- By eliminating the storage process, retail chains can significantly reduce costs of distribution - reduces the expenditure on handling and storage, minimizes the amount of product downtime on the way from production to its final destination	- cross docking requires accurate synchronization of all processes of goods receiving and delivering - Sharing the information, a reliable communication and ensuring the quality and quantity of products received from suppliers determine the efficiency of the system - Therefore, there is a need for information technology simplifying the procedures for orders placement and execution - Failure in meeting the delivery date by the manufacturer is either a delivery rejection (failing) or a long wait of a driver for a free time block when the supply may possibly be discharged - The acceptance of the supplies of both goods and documents are subjected to a thorough control by the network staff. Any breach of the delivered documentation, goods, packaging, distribution of products, results in rejection of the delivered goods - noticeable increase in the cost of information technology and telecommunication solutions, training for employees - a large number of shipments and fast transshipment process may also lead to an increased risk of errors	Relevant remarks both in the case study as in literature sources
14	Literature review, guidelines for successful use and implementation	- Consolidation of shipments - Cross-docking however is an approach that eliminates the two most expensive handling operations: storage and order picking - Cross-docking corresponds with the goals of lean supply chain management: smaller volumes of more visible inventories that are delivered faster and more frequently - cost reduction (warehousing costs, inventory-holding costs, handling costs, labor costs, transportation costs) - shorter delivery lead time	- Requires a correct synchronization of incoming (inbound) and outgoing (outbound) vehicles. However, a perfect synchronization is difficult to achieve - Hardware for a cross-docking system (material handling devices, sorting systems, etc.) might come off the shelf and is easily available today. But the software needs to be tailored to the F requirements and is in general relatively less developed, although it is as important as hardware to cross-docking success - Compared with regular distribution, the information flow to support cross-docking is significantly more important	Relevant.

		- improved customer service - reduction of storage space - faster inventory turnover - fewer overstocks - reduced risk for loss and damage - improved resource utilization (e.g. full truckloads) - better match between shipment quantities and actual demand		
15	Case study: situation awareness to respond to operational dynamics	- obtaining economies in transportation costs by consolidation of multiple smaller-sized loads to full truck loads	- Scheduling procedures to synchronize all transport and transshipment operations in the transportation network. - The operational performance at cross docks is highly sensitive to dynamics in the environment - the operational performance at cross docks is greatly influenced by the real-time decisions made by the dispatchers in order to control on-going transshipment operations - The complex and highly dynamic operating environment of cross docks make that unexpected events may occur frequently - Truck delays can have a negative impact the performance of the cross docking operations by creating additional waiting times - Unexpected peaks of pallet numbers can have a negative impact on operational performance	
16	dynamic design methodology to select control policies and deter-mine layout rules for cross docking facilities	- reduction of waiting time in the warehouse	- planning issues that exist are complex	Article is about lay out of the storage area and not very relevant in our context
17	Literature review: concepts, settings, decision problems	- consolidation of many smaller shipments between multiple shippers and recipients, so that only full truckloads are transported, results in lower transportation costs - recipients profit from a reduced number of truck deliveries, which relieves their receiving areas - final value creation can be fulfilled and a delayed product differentiation (postponement) can be realized	- any additional stop causes double handling and increases delivery times - The consolidation process within a cross-dock causes additional variable and fixed costs for staff and resources, and jeopardizes timely deliveries to final customers - efficient transshipment processes are required where inbound and outbound truckloads are synchronized - multiple interdependent decision problems need to be solved that need a holistic approach - Only if the customers accept the longer delivery times will the additional processing time for double handling in the cross-docking terminal (compared with a point-to-point delivery) seem acceptable	Relevant
18	Literature review: revealing potential research areas, three were	- Cross-docks can add value to supply chains where potential exists to improve transport efficiency, reduce inventory, or	- A cross-dock does not operate in isolation and therefore cannot be optimized independently from the upstream and downstream processes	Relevant assessment of general classifications

	chosen for detailed analysis	speed movement of products - cost savings from combining loads	- Up- and downstream visibility Nine criteria are listed: 1. Appropriate products (similar handling characteristics, consistent movement thus continual loading, speed of movement) - 2. Understanding how cross-dock-based supply chains work for different types of cross dock - 3. Effective computer systems (with order management, advanced shipping notice, yard management system, cross dock management system, track and trace across the supply chain) - 4. Efficient physical facility design and layout - 5. Process improvement and problem-solving capability in the cross-dock - 6. Reliable product suppliers - 7. Specialist and reliable supply chain service providers - 8. Uniquely skilled management and staff - 9. Work balancing and minimization (consistent throughout the day) - downstream customers should require, and be able to receive continually, significant quantities of products at all times for desired speed of movement - there is neither the time nor the space to correct significant numbers of errors	and operational criteria. different countries and industry segments investigated.
19	Comparison with warehouse, design parameters, practical design methods	- There is less inventory carried in the supply chain for the cross-dock operation, and the building is less costly as it is a simple, low-roofed facility when compared to a warehouse or distribution center	- High personnel capability, high facility design effectiveness and high systems capability is needed for potential highly effective and efficient crossdocking operations	Overlapping with 19 and design consideration not so relevant for this review
20	Case study: main steps in designing a cross docking project	- eliminates the inventory-holding steps of a warehouse while it allows to serve the consolidation and shipping functions - reduction of noise pollution, road accident and urban blight		Not relevant. Case study is specific and hardly about crossdocking
21	Two models are developed comparing crossdocking and warehouse strategy, case study to illustrate the models developed	- reduce inventory and improve responsiveness to various customer demands - reduce transportation cost and delivery time without increasing inventory - achieve the economy of scale that comes with purchasing full truckloads of products while avoiding the usual inventory and handling costs		A mathematical model is described. Though, because it is a comparison between crossdocking and warehousing it is still relevant

		- reduction of order cycle time, thereby improving the flexibility and responsiveness of the distribution network		
22	Literature review, case example, ILP model	- reduce the time inventory spends in the supply chain, and thus inventory costs - potential to eliminate both storing and picking - less space, equipment, and labor required for handling and storing the products, as well as a reduced risk of product damages and obsolescence	- proper planning and management tools are necessary in order to realize the benefits of cross-docking - mean demand relative to TL capacity should be a primary consideration when considering the use of cross-docking	A mathematical model is described. Though, because it is strategical oriented it is still relevant
23	Case study: outsourcing decision for crossdocking of two automotive firms	- facilitate the efficient management and operation of a supply chain in support of lean manufacturing	- Significant skills are needed to coordinate the many suppliers, the various inbound materials, and the material requirements of the various assembly plants that a cross dock serves - Sophisticated technological systems and processes are needed to optimize transportation routes, maximize the volume of material on trucks, adapt to unplanned events, and ensure that material flows efficiently through the cross dock in a manner that maximizes quality, minimizes cost, and is timely - a management view that perceives distribution and logistics as a commodity can result in a less integrated structure, and thus, poorer performance	Article is mainly about lean and outsourcing and not so much about crossdocking
24	Short article that describes requirements for successful crossdocking	- eliminating storing and picking, the two most expensive warehousing operations - increases inventory turns, thus reducing inventory carrying costs and speeding the flow of product to the consumer	- Partnering with other members of the distribution chain is necessary - Absolute confidence in the quality and availability of product - Communications between supply chain members, information must be immediately available - Communications and control within the cross docking operation requires a proper WMS suited for crossdocking - greatly reduces the flexibility to level workload - Someone responsible for Tactical management is of vital importance (Door allocation, resources, separate from first line supervisor)	Relevant assessment of criteria for successful cross dock implementation
25	General classification and case study about crossdocking	- stock reduction, fixed resource reduction and more responsive operating systems - turning expensive delivery consignments into economic loads through consolidation and resource sharing - increased stock flow, reduced stockholding, improved resource utilization and reduced delivery lead times		Relevant article on general considerations

		- reduced warehousing property/area		

Appendix D: Interviews

Table 13 and Table 14 list the interview respondents and the factors they touched upon during their interviews. After these tables, the 19 factors derived from the interviews, each with an explanation, are listed.

Table 13 Interview respondents

Respondent	Function
1. Rensa	Commercial director
2. Rensa	Logistics director
3. Rensa	Head of transport
4. Rensa	Head of warehouses
5. Rensa	Head of logistics management
6. Rensa	Controller
7. Hospitality wholesaler (external)	Transportmanager
8. Solar energy company (external)	Transport- and warehousemanager
9. Logistics service provider (external)	Facility manager

Table 14 Outcomes per respondent

Factor	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1. Rensa		x			x				x	x		x	x	x		x	x		
2. Rensa	x	x		x	x	x	x	x	x	x	x	x	x	x	x	x			
3. Rensa	x	x		x		x	x		x			x	x	x		x	x		
4. Rensa	x	x						x	x	x		x	x	x		x	x		
5. Rensa	x	x	x		x	x			x			x	x	x	x	x			x
6. Rensa	x	x		x			x		x	x		x	x	x		x	x		x
7. Hospitality wholesaler (external)	x			x		x			x			x		x		x		x	
8. Solar energy company (external)	x			x									x						
9. Logistics service provider (external)	x					x	x		x			x	x	x		x			

1. More flexible and efficient use of trucks from hub

From hubs, the average distance to customers becomes smaller. This makes it possible for delivery trucks to come back to the hub to load a truck twice or to do an extra trip when the initial delivery round is finished earlier. Currently, truck drivers often finish their delivery earlier, not utilizing their 8 or 9 working hours, or they do not have their truck fully loaded because their working hours are the limiting factor. From a hub the truck utilization rate and truck driver hours utilization can possibly increase. In general, the expectation is that total transport costs will decrease in a new logistics network.

2. Effect of hub capacity on the effectivity and efficiency of an organization on site

The size of a hub influences the effectivity and efficiency of an organization on site. A very small organization is very vulnerable, for instance for demand fluctuations and employee illness. A very large hub on the other side automatically creates a larger sales area with larger average distance to customers. There are some important concerns regarding the organization on a hub. Which functions do we want to place on a hub? This ranges from a minimum number of people who only load and unload, to an extensive organization with an office and someone from service and planning on site. Another concern is the bond between employees at the hub and the central organization of Rensa. How can this bond be maintained? What may be an option on the hubs is to let drivers load and unload their own trucks. Another idea is to outsource the hub itself, while keeping transport to and from a hub inhouse. Another concern are the peak times in work at the hubs. If this peak concentrates at night, it may be difficult to find (good) people wo who want to work at a hub.

3. Effect on transport planning

If we want to be able to shuttle between central DCs and hubs throughout the day, before the transport planning is complete, we need to break up the Netherlands in a number of fixed smaller postal code areas tied to a hub. This takes away flexibility in transport planning which may reduce efficiency of delivery routes.

4. More variable demand pattern on hub due to smaller scale

There are fewer customers and less sales served from a hub compared to the current central organization. The impact of individual customers and projects is relatively bigger, possibly leading to increased demand fluctuations.

5. Risk of errors

Part of the products go through an extra handling step in a new network design. This increases the change on errors. Also, in Rensa’s central DCs a product damaged during loading can quickly be replaced from stock, something that is not possible on a hub anymore. The other side is that handling at a hub takes place on smaller scale and is less complex compared to the current operation.

6. Deliver project inventory from hub

Crossdocking hubs will not hold regular inventory. What may be possible is that they store project inventory: products for bigger projects, that are already allocated to a sales order. These projects are often planned and known weeks upfront. Actual allocation of inventory to a sales order now takes place 10 days before the delivery date by standard. Some orders are called for in small batches by the customer. On the hub these small batches are closer to the customer and do not use the limited space available in the central DCs. When inventory is ordered for a specific customer, suppliers may deliver to the hub directly, skipping a step in the supply chain. A question is how suppliers look upon delivering to hubs, as they are used to delivering all goods at one location. As there is no time pressure on delivering project orders to a hub, this may be done on idle moments to balance workload at the central DCs. These orders may then also be used to fill less-than-full truckloads, preventing inefficient trips. Project orders often consist of many and voluminous items. When delivering further away from Doetinchem, these orders are difficult to fit in the box truck concept that Rensa uses most. With a smaller distance to customers from a hub, this may become easier.

7. Network better suitable for in-house electrical (urban) distribution

Rensa has the ambition to be CO2 neutral in 2030. Electrical distribution is a vital part of this goal. Electrical vehicles are limited in their range and distributing the Netherlands with electrical vehicles from Doetinchem is currently infeasible. Due to the shorter average distance to customers, this will be easier from a hub. Electrification may also enable Rensa to keep delivering in urban areas. Restrictions on weight and pollution are becoming stricter and electric trucks may get the exceptions to still be allowed to deliver in city centers.

8. Handling

The amount of handling changes in different parts of the logistics network in a new network design. More handling arises due to additional overloading at hubs; however, the operation is smaller and less complex. At the central DCs handling may be reduced because the central DCs can all directly deliver to the hubs. The trucks that transport from the central DCs to hubs can be loaded unsorted, where delivery trucks departing from DC4 have to be loaded consolidated per stop and on stop order. This can save time and shifts work away from DC4 to the hubs. If we want to be able to shuttle between central DCs and hubs during the day, the loading process and loading times at the central DCs will change. Only loading at the end of the day is no longer possible. Although change is needed, it may lead to a more even work distribution at DC4 particularly. Meeting loading and delivery deadlines may become easier.

9. Impact of a more decentralized network on information flows and IT

The question arises if the current software and IT systems are still suitable in a new logistics network. A more decentralized network may ask for increased goods traceability to prevent errors and solve them quickly. There is no standard stock on the hubs. However, having project inventory may also have impact on the IT systems and order processing at the hub. The availability of a form of Business Intelligence and predictive ability may also become more important. At this moment Rensa is well known with the effects of the environment on the logistics network and a lot of knowledge is obtained by operating the network over the years. In a new network predicting the impact of for instance customer demand may become more important.

10. Night distribution via hubs

At this moment night distribution is a separate branch of delivery. Delivery is outsourced, be it with long-term relationships, and truck pitches are not used for night delivery. In the new situation, night distribution can also go via the hubs. In this case the night orders can be shuttled to the hubs together with day orders, throughout the day.

11. Truck fleet composition and delivery

The question will arise if the current truck fleet composition is still satisfactory in a new situation. A decision should be made on the type of truck that will deliver on which route to be as efficient as possible and minimize costs. Delivery from central DCs to hubs will probably be done in larger trucks with utilization as primary focus, while keeping complexity as low as possible. Delivery from hub to customer will probably be done with box trucks and vans, as is also done in the current situation.

12. (Partially) combining counter function and hub

The hubs may be partially combined with the currently existing counters throughout the Netherlands. The counters function as point of knowledge and first line stock for customers. The hub can take over the function of a counter, but this would mean that a number of items must be kept in stock at the hub. The other way around may also be possible. In this case the counter needs additional warehousing capacity to function as a hub.

13. Return flow through hubs

The return flow of goods changes when hubs are added to the logistics network. These returned goods must come back to Doetinchem, which requires handling at the hubs. New processes must be designed for this.

14. Construction site logistics

Construction site logistics is a service that Rensa delivers for bigger building projects. Goods are not delivered in one batch to the building site, but are for instance delivered per floor level at a time and are packed together per apartment. These deliveries are often done at night, avoiding the crowded building site during working hours. Also, night delivery makes it possible for installers to start working immediately when they arrive at the building site. When the service is combined with prefabrication, it makes the job for installers even easier. At this moment it is difficult to carry out construction site logistics far away from Doetinchem. Moving the goods and people from Doetinchem to the building site is not economically viable and therefore moving companies are often hired in the current situation. With hubs, there is a bigger organization closer to the construction site, which may open new possibilities.

15. *Put the sales order, awaiting to be called off, already on the hub*

This factor is related to the project inventory factor. The difference with the factor about delivering project inventory, is that here we concern regular orders that are of smaller size. Smaller customers often work on a project basis for themselves, which means that they know their orders a few days in advance. Sometimes these customers want to work on a different project on a given day due to circumstances like a customer not being present. If orders for next days are already transported to a hub, these customers may be able to pick up or receive their products for another project on the same day to continue work. These orders are within the ten day range and are thus already allocated to a sales order. They follow the normal flow via the central DCs always, which is another difference with project inventory in the previous point. Prefab orders may also apply for this earlier delivery to the hubs.

16. Effect on people in existing organization

In a new logistics network some functions may be removed in Doetinchem. The biggest group will be drivers who are stationed in Doetinchem. The number of drivers stationed in Doetinchem will decrease, as more delivery takes place from the hubs. The same holds for drivers now stationed at the truck pitches.

17. Same day delivery

From a hub, it may be possible to set up processes in such a way that a form of same day delivery can be done. With order words, ordering today would mean being delivered today. The question is if there is a demand for this from the customer. Also, customers of Rensa only have benefit from this service if they also have time left on a day to work with the delivered products. This means that products cannot be delivered late in the afternoon.

18. Return of waste from customers

Some customers have the wish that returning of waste materials is also organized by Rensa. With hubs these flows do not have to go all the way to Doetinchem and may be organized easier.

19. Direct delivery from regional supplier to hub nearby

Some suppliers are located in different parts of the Netherlands, far from Doetinchem. These suppliers may prefer to deliver to a hub nearby instead of delivering to Doetinchem or Didam. However, generally a supplier will still have to deliver goods to one of the central DCs of Rensa too.

Appendix E: Initializing, running and using Docker images

After having mounted a host-directory in the Docker container, one can initialize the OSRM backend service. The following commands are run in Windows PowerShell when Docker is already running:

docker run -t -v data osrm/osrm-backend:latest osrm-extract -p /opt/car.lua /data/netherlands-latest.osm.pbf

docker run -t -v ${pwd}:/data osrm/osrm-backend:latest osrm-contract /data/netherlands-latest.osrm

The first command line extracts the map. This can be very memory intensive. OSRM supports different transport modes, which can for instance be bike, car or foot. There is *car.lua* in the command line because we want to use the car profile. The second command line prepares Contraction Hierarchies, a speed-up technique for the routing algorithm.

We can now run our docker image, which will also activate our routing server so that it is ready to use. This is done using the following command:

docker run -t -i -p 5000:5000 -v ${pwd}:/data osrm/osrm-backend:latest osrm-routed –max-table-size=500000 /data/netherlands-latest.osrm

We give an extra optional argument in the command line for the max table size. This refers to the maximal number of values that can be in a calculated durations matrix. With DC4, 12 candidate locations and 33484 customers we need a table size of at least 13 x 33484, which is sufficed with 500.000.

When the command is ran successfully, the Window PowerShell prompt returns: 'running and waiting for request', and our local OSRM server is ready to be used.

Using OSRM

As said before, OSRM was originally built in C++. Here, the routing machine could be called using the 'curl' command. In Python there is no curl command, so we use the request package as alternative. This package is used to send HTTP request. After having installed and imported the requests package, we can send a request using the command:

requests.get(('http://localhost:5000/table/v1/driving/coordinates?option=()')

The URL sends a request to our local server, in this case calling the durations table service. At the place of coordinates, a string of coordinates can be given. Extra options, like which coordinates are sources or which coordinates are destinations, can be given at the end of the command.

Appendix F: Python code: location model

```python
#%%Create and fill durations matrix,this only works with OSRM running


def DurationsMatrix():

    #import openpyxl for Excel support
    import openpyxl  # Seems unused but is needed for Excel files
    #Import pandas package for Excel import
    import pandas as pd
    #import request to call the routing server
    import requests

    #Import coordinates from Excel

    #List with coordinates is formatted as follows:
    #0:DC4Doetinchem,1:HubLibra,2-12:OtherHubs,13-end:Customers
    CoordinateTable = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - Data2020.xlsx',
                                    sheet_name='DurationsMatrix')
    global CoordinateList  # Make the variable global for use in other procedures
    CoordinateList = CoordinateTable.values.tolist()  # Convert the Excel DataFrame to a list
    CoordinateString = str(CoordinateList)  # Convert the list to a string for the request url
    #Format coordinatelist from Excel for the URL request to osrm
    CoordinateString = ''.join(CoordinateString).replace('[', '')
    CoordinateString = ''.join(CoordinateString).replace('],', ';')
    CoordinateString = ''.join(CoordinateString).replace(' ', '')
    CoordinateString = ''.join(CoordinateString).replace(']', '')
    #Osrm request to get a distance matrix
    OsrmResponse = requests.get('http://localhost:5000/table/v1/driving/{}?sources=0;1;2;3;4;5;6;7;8;9;10;11;12'
                                .format(CoordinateString))
    ResponseDictionary = OsrmResponse.json()  # Get dictionary with the response content from the osrm request
    DurationsMatrix = ResponseDictionary['durations']  # Get the durations matrix durations[i][j] from the dictionary
    return DurationsMatrix

#%%Define the duration of driving from DCs to hubs and add durations to DurationsMatrix


def ShuttleDistances(DurationsMatrix):

    #Import pandas package for Excel import
    import pandas as pd
    CandidateLocations = range(1, 13)  # List for 12 candidate locations

    CostBakwagenHour = 53.75  # All in costs per hour bakwagen
    CapacityBakwagen = 6000  # Capacity of Bakwagen in VolumeGewicht
    CostBakwagenHourVolumeGewicht = CostBakwagenHour / CapacityBakwagen  # Cost Bakwagen per hour per VolumeGewicht eenheic

    CostLzvHour = 78.74  # All in costs per hour Lzv
    CapacityLzv = 18000  # Capacity of Lzv in VolumeGewicht
    CostLzvHourVolumeGewicht = CostLzvHour / CapacityLzv  # Cost Lzv per hour per VolumeGewicht eenheid

    global DurationsMatrix2  # Make the variable global for use in other procedures
    DurationsMatrix2 = DurationsMatrix  # Create a copy of DurationsMatrix

    # Iterate over hubs and define duration based on the ratio between Bakwagen and Lzv costs
    for hub in CandidateLocations:
        ShuttleDuration = DurationsMatrix[0][hub] * CostLzvHourVolumeGewicht / CostBakwagenHourVolumeGewicht
        for Customer in range(len(CoordinateList)):  # Loop over all customers
            # Add shuttleduration to hub-customer duration
            DurationsMatrix2[hub][Customer] = DurationsMatrix2[hub][Customer] + ShuttleDuration

    #Export the DurationsMatrix to Excel
    DurationsDataFrame = pd.DataFrame.from_dict(DurationsMatrix2)  # List to Dataframe for Excel export
    TransposedDataFrame = DurationsDataFrame.transpose()  # Transpose rows and columns
    TransposedDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\Python - DurationsMatrix.xlsx',
                                 header=False, index=False)  # Export
    return()

#%%Run the above functions


ShuttleDistances(DurationsMatrix())  # Call the DurationsMatrix functions and run ShuttleDistances with its result
```

Figure 25 OSRMDurationsMatrix module

```python
#%%Without OSRM the distance matrix can still be accessed using this Excel file


def DurationsFromExcel():

    #Import pandas package for Excel import
    import pandas as pd

    #Read duration data from Excel
    DurationsDataFrame = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\Python - DurationsMatrix.xlsx', header=None)
    TransposedDurationsDataFrame = DurationsDataFrame.transpose()  # Transpose because columns and rows are interchanged
    global DurationsMatrix2  # Make the variable global for use in other procedures
    DurationsMatrix2 = TransposedDurationsDataFrame.values.tolist()  # Convert DataFrame to list
    return()

#%% Create and fill VolumeGewicht list


def VolumeGewicht():

    #Import pandas package for Excel import
    import pandas as pd

    global VolumeGewichtList  # Make the variable global for use in other procedures
    #Import VolumeGewicht from Excel
    VolumeGewichtTable = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - Data2020.xlsx',
                                       sheet_name='VolumeGewicht')
    VolumeGewichtList = VolumeGewichtTable.values.tolist()  # Convert the Excel DataFrame to a list

#%% Iterate over configuration and store open hubs


def LocationModel(NumberOfHubs):

    #Import pandas package for Excel import
    import pandas as pd

    DurationPerConfiguration = {}  # Define dictionary for weighted configuration durations
    if NumberOfHubs > 12:  # There are only 12 candidate locations
        raise Exception('NumberOfHubs not more than 12')  # An error is thrown
    elif NumberOfHubs > 0:  # One or more hubs are given as input
        # import itertools to make hub combinations list
        import itertools
        # make list with unique hub configurations
        CandidateLocations = range(1, 13)  # List for 12 candidate locations
        #Itertools creates list of unique combinations
        ConfigurationsList = list(itertools.combinations(CandidateLocations, NumberOfHubs))
        for ConfigurationNr in range(len(ConfigurationsList)):  # iterate over configurations
            ConfigurationWeighedDuration = 0  # initialize
            Configuration = ConfigurationsList[ConfigurationNr]  # Store the configuration so we can iterate over the hubs in it
            for Customer in range(13, len(VolumeGewichtList)):  # iterate over all customers
                BestDuration = DurationsMatrix2[0][Customer]  # initialize the best duration with the duration to DC4(Doetinchem)
                for hub in Configuration:  # iterate over hubs in configuration
                    Duration = DurationsMatrix2[hub][Customer]  # Duration hub-customer
                    if Duration < BestDuration:  # if this duration is smaller than the current best for this customer
                        BestDuration = Duration  # new BestDuration
                # Update total configuration duration
                ConfigurationWeighedDuration = ConfigurationWeighedDuration + BestDuration * VolumeGewichtList[Customer][0]
            # Store this configurations duration with the configuration as index
            DurationPerConfiguration[Configuration] = ConfigurationWeighedDuration
    else:  # 0 hubs so current situation with only DC4
        ConfigurationWeighedDuration = 0  # initialize
        for Customer in range(13, len(VolumeGewichtList)):  # iterate over all customers
            # Add each customers weighted duration to the total duration
            ConfigurationWeighedDuration = ConfigurationWeighedDuration + DurationsMatrix2[0][Customer] * VolumeGewichtList[Customer][0]
        DurationPerConfiguration['NoHubs'] = ConfigurationWeighedDuration  # Store duration with 'NoHubs' as index
    ConfigurationDurationDataFrame = pd.DataFrame.from_dict(DurationPerConfiguration, orient='index')  # Dictionary to DataFrame
    ConfigurationDurationDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - Output.xlsx')  # Export
    return()

#%%Code where the needed data for the LocationModel is gathered, RUN THIS ONCE SO THAT DURATIONSMATRIX2 AND VOLUMEGEWICHTLIST ARE FILLED


DurationsFromExcel()  # Call DurationsFromExcel function
VolumeGewicht()  # Call VolumeGewicht function

#%%Code that actual calls the location model functionality, MAKE SURE THAT THE CODE ABOVE IS RUN ONCE SO THAT DURATIONSMATRIX2 AND
#VOLUMEGEWICHTLIST ARE DEFINED AND FILLED

#GIVE THE NUMBER OF HUBS FOR WHICH YOU WANT TO DO CALCULATIONS HERE
NumberOfHubs = 0  # Define the number of hubs for which the model is run
LocationModel(NumberOfHubs)  # Call LocationModel function with NumberOfHubs as input parameter
```

Figure 26 LocationModel module (1)

```python
#%%Retrieve customer allocation of desired configuration


def CustomerAllocation(Configuration):

    #Import pandas package for Excel import
    import pandas as pd

    #Create dictionaries to save customer allocation and durations
    CustomerToHubAllocation = {}
    AllocationDuration = {}

    for Customer in range(13, len(VolumeGewichtList)):  # iterate over all customers
        BestDuration = DurationsMatrix2[0][Customer]  # initialize the best duration with the duration to DC4(Doetinchem)
        BestHub = 0  # initialize best hub as 0, which is DC4(Doetinchem)
        if type(Configuration) == tuple:
            for hub in Configuration:  # iterate over hubs in configuration
                Duration = DurationsMatrix2[hub][Customer]  # duration from current hub to current customer
                if Duration < BestDuration:  # if this duration is smaller than current best
                    BestDuration = Duration  # make current best
                    BestHub = hub  # save best hub for this customer
        else:
            Duration = DurationsMatrix2[Configuration][Customer]  # duration from current hub to current customer
            if Duration < BestDuration:  # if this duration is smaller than current best
                BestDuration = Duration  # make current best
                BestHub = Configuration  # save best hub for this customer
        CustomerToHubAllocation[Customer] = BestHub  # Store best allocated hub with this customer
        AllocationDuration[Customer] = BestDuration  # Store best duration with this customer
    #Excel export
    CustomerAllocationDataFrame = pd.DataFrame.from_dict(CustomerToHubAllocation, orient='index')  # Dictionary to DataFrame
    CustomerAllocationDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - CustomerAllocation.xlsx')  # Export

    AllocationDurationDataFrame = pd.DataFrame.from_dict(AllocationDuration, orient='index')  # Dictionary to DataFrame
    AllocationDurationDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - CustomerDuration.xlsx')  # Export
    return()

#%%Code that calls the CustomerAllocation procedure, USE THIS ONE WHEN DURATIONSMATRIX2 AND VOLUMEGEWICHTLIST ARE ALREADY DEFINED


#GIVE THE DESIRED CONFIGURATION IN FORMAT (hubnr, hubnr, etc) HERE
#1:HubLibra,2-12:OtherHubs
Configuration = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)  # Desired configuration

CustomerAllocation(Configuration)  # Call CustomerAllocation procedure

#%%Code that calls the CustomerAllocation procedure, USE THIS ONE WHEN DURATIONSMATRIX2 AND VOLUMEGEWICHTLIST ARE NOT DEFINED


#GIVE THE DESIRED CONFIGURATION IN FORMAT (hubnr, hubnr, etc) HERE
#1:HubLibra,2-12:OtherHubs
Configuration = (4, 7, 12)  # Desired configuration

#Import DurationsMatrix2 from Excel calling the DurationsFromExcel() procedure
DurationsFromExcel()  # Call DurationsFromExcel function
VolumeGewicht()  # Call VolumeGewicht function
CustomerAllocation(Configuration)  # Call CustomerAllocation procedure

```

Figure 27 LocationModel module (2)

```python
#%%import coordinates from Excel


import pandas as pd
import matplotlib.pyplot as plt

#Import customer coordinates and desired customer allocation
#List with coordinates is formatted as follows:
#0:DC4Doetinchem,1:HubLibra,2-12:OtherHubs,13-end:Customers
CoordinateDataFrame = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - Data2020.xlsx',
                                    sheet_name='DurationsMatrix', skiprows=14, header=None)

#%%Plot on map

AllocationDataFrame = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\LocationModel - Resultaten.xlsx',
                                    sheet_name='5 Hubs', usecols='E')

#Define a DataFrame for each hub to store allocated customers in
DC4DataFrame = pd.DataFrame()
Hub1DataFrame = pd.DataFrame()
Hub2DataFrame = pd.DataFrame()
Hub3DataFrame = pd.DataFrame()
Hub4DataFrame = pd.DataFrame()
Hub5DataFrame = pd.DataFrame()
Hub6DataFrame = pd.DataFrame()
Hub7DataFrame = pd.DataFrame()
Hub8DataFrame = pd.DataFrame()
Hub9DataFrame = pd.DataFrame()
Hub10DataFrame = pd.DataFrame()
Hub11DataFrame = pd.DataFrame()
Hub12DataFrame = pd.DataFrame()

#Append each hub with the customers that are allocated to it
for Customer in range(len(AllocationDataFrame)):
    if AllocationDataFrame.iloc[Customer][0] == 0:
        DC4DataFrame = DC4DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 1:
        Hub1DataFrame = Hub1DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 2:
        Hub2DataFrame = Hub2DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 3:
        Hub3DataFrame = Hub3DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 4:
        Hub4DataFrame = Hub4DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 5:
        Hub5DataFrame = Hub5DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 6:
        Hub6DataFrame = Hub6DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 7:
        Hub7DataFrame = Hub7DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 8:
        Hub8DataFrame = Hub8DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 9:
        Hub9DataFrame = Hub9DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 10:
        Hub10DataFrame = Hub10DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 11:
        Hub11DataFrame = Hub11DataFrame.append(CoordinateDataFrame.iloc[Customer])
    elif AllocationDataFrame.iloc[Customer][0] == 12:
        Hub12DataFrame = Hub12DataFrame.append(CoordinateDataFrame.iloc[Customer])

#Change the column names of each DataFrame that is not empty
if not DC4DataFrame.empty:
    DC4DataFrame.columns = ['Lon', 'Lat']
if not Hub1DataFrame.empty:
    Hub1DataFrame.columns = ['Lon', 'Lat']
if not Hub2DataFrame.empty:
    Hub2DataFrame.columns = ['Lon', 'Lat']
if not Hub3DataFrame.empty:
    Hub3DataFrame.columns = ['Lon', 'Lat']
if not Hub4DataFrame.empty:
    Hub4DataFrame.columns = ['Lon', 'Lat']
if not Hub5DataFrame.empty:
    Hub5DataFrame.columns = ['Lon', 'Lat']
if not Hub6DataFrame.empty:
    Hub6DataFrame.columns = ['Lon', 'Lat']
if not Hub7DataFrame.empty:
    Hub7DataFrame.columns = ['Lon', 'Lat']
if not Hub8DataFrame.empty:
    Hub8DataFrame.columns = ['Lon', 'Lat']
if not Hub9DataFrame.empty:
    Hub9DataFrame.columns = ['Lon', 'Lat']
```

Figure 28 PlotOnMap module (1)

```python
if not Hub10DataFrame.empty:
    Hub10DataFrame.columns = ['Lon', 'Lat']
if not Hub11DataFrame.empty:
    Hub11DataFrame.columns = ['Lon', 'Lat']
if not Hub12DataFrame.empty:
    Hub12DataFrame.columns = ['Lon', 'Lat']


MapBoundaries = (2.5, 7.69, 50.70, 53.455)  # Define map boundaries
NetherlandsMap = plt.imread(r'C:\Users\User\Documents\6. Master Thesis\NetherlandsMap.png')  # read Netherlands map
fig, ax = plt.subplots(figsize=(12, 12))  # define plot

#Plot desired DataFrames
ax.scatter(DC4DataFrame.Lon, DC4DataFrame.Lat, s=1, alpha=1, c='b')
ax.scatter(Hub2DataFrame.Lon, Hub2DataFrame.Lat, s=1, alpha=1, c='r')
ax.scatter(Hub3DataFrame.Lon, Hub3DataFrame.Lat, s=1, alpha=1, c='Purple')
#ax.scatter(Hub4DataFrame.Lon, Hub4DataFrame.Lat, s=1, alpha=1, c='cyan')
ax.scatter(Hub7DataFrame.Lon, Hub7DataFrame.Lat, s=1, alpha=1, c='g')
ax.scatter(Hub8DataFrame.Lon, Hub8DataFrame.Lat, s=1, alpha=1, c='Brown')
#ax.scatter(Hub9DataFrame.Lon, Hub9DataFrame.Lat, s=1, alpha=1, c='grey')
ax.scatter(Hub12DataFrame.Lon, Hub12DataFrame.Lat, s=1, alpha=1, c='yellow')


#Plot the desired hub locations
ax.scatter(6.292126, 51.948545, s=80, alpha=1, c='orange')  # DC4
#ax.scatter(4.640377, 52.467234, s=80 alpha=1, c='orange')  # Hub1
ax.scatter(6.55738, 52.972404, s=80, alpha=1, c='orange')  # Hub2
ax.scatter(5.954182, 52.93043, s=80, alpha=1, c='orange')  # Hub3
#ax.scatter(6.199197, 52.709508, s=80, alpha=1, c='orange')  # Hub4
#ax.scatter(4.883822, 52.213386, s=80, alpha=1, c='orange')  # Hub5
#ax.scatter(5.428043, 52.167213, s=80, alpha=1, c='orange')  # Hub6
ax.scatter(5.122727, 52.0525, s=80, alpha=1, c='orange')  # Hub7
ax.scatter(4.747329, 52.071409, s=80, alpha=1, c='orange')  # Hub8
#ax.scatter(4.999241, 51.84177, s=80, alpha=1, c='orange')  # Hub9
#ax.scatter(5.433582, 51.904416, s=80, alpha=1, c='orange')  # Hub10
#ax.scatter(4.82704, 51.67349, s=80, alpha=1, c='orange')  # Hub11
ax.scatter(5.396976, 51.48738, s=80, alpha=1, c='orange')  # Hub12

ax.set_title('Customer division over hubs')  # Change plot title
ax.set_xlim(MapBoundaries[0], MapBoundaries[1])  # Set graph boundaries to match the Netherlands map
ax.set_ylim(MapBoundaries[2], MapBoundaries[3])
ax.imshow(NetherlandsMap, zorder=0, extent=MapBoundaries, aspect='equal')  # show the graph and map
```

Figure 29 PlotOnMap module (2)

Appendix G: Python code: routing model

```python
def RoutingLogic(Configuration, VolumetricWeightArray, ServiceTimeArray, DurationsMatrix, HubDurationsMatrix):

    #import openpyxl for Excel support
    import openpyxl  # Seems unused but is needed for Excel files
    #Import pandas package for Excel import
    import pandas as pd

    #Initialization
    VehicleCapacity = 6000  # Box truck capacity
    DriverWorkTime = 31500  # Truck drivers work 9 hours (32400 seconds)
    AllRoutes = {}  # Dictionary to store routes of all hubs
    AllRouteMetrics = {}  # Dictionary to store all routemetrics of all hubs

    #Loop over the hubs in the configuration
    for hub in Configuration:
        #Create a list with all customer for current hub
        CustomerHubs = pd.read_excel(r'C:\Users\User\Documents\6. Master Thesis\Data 2020 - November.xlsx',
                                     sheet_name='Maandag', usecols="R")
        CustomerHubList = []
        for c in range(len(CustomerHubs)):  # loop over hub column in Excel
            if (CustomerHubs.iloc[c, 0] == hub):  # if customer belongs to current hub
                CustomerHubList.append(c)  # Store customer belonging to current hub

        #Calculate savings
        SavingsDictionary = {}  # Create dictionary for savings
        for customerfrom in range(len(CustomerHubList)):  # loop over customers of this hub, these are customer i in SavingsArray(i,j)
            for customerto in range(customerfrom + 1, len(CustomerHubList)):  # customer j in SavingsDictionary(i,j)
                SavingsDictionary[customerfrom, customerto] = (HubDurationsMatrix[hub][CustomerHubList[customerfrom] + 13] +
                                                               HubDurationsMatrix[hub][CustomerHubList[customerto] + 13] -
                                                               DurationsMatrix[CustomerHubList[customerfrom]][CustomerHubList[customerto]])

        #Sort the savings array in descending order
        SortedSavingsList = sorted(SavingsDictionary.items(), key=lambda x: x[1], reverse=True)  # Sort savings and store in list

        #Built routes
        Routes = []  # Create dictionary for rouRoutes[CustomerInRoute[SortedSavingsList[entry][0][0]]]tes
        CustomerInRoute = {}  # Create dictionary that stores the assigned route per customer
        RouteMetrics = {}  # Create dictionary to stores routes duration and volumetricweight
        for entry in range(len(SortedSavingsList)):  # loop over the SortedSavingsList

            #Both locations already in route
            if ((SortedSavingsList[entry][0][0] in CustomerInRoute) and (SortedSavingsList[entry][0][1] in CustomerInRoute)):
                #if A and B are in the same route we can skip
                if not (CustomerInRoute[SortedSavingsList[entry][0][0]] == CustomerInRoute[SortedSavingsList[entry][0][1]]):
                    #Check if both A and B are on the outside of their routes
                    if (((Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][0] == SortedSavingsList[entry][0][0]) or
                        (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][-1] == SortedSavingsList[entry][0][0])) and
                         ((Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][0] == SortedSavingsList[entry][0][1]) or
                          (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][-1] == SortedSavingsList[entry][0][1]))):
                        #RouteDurationWithService = DurationWithServiceA + DurationWithServiceB - SavingsAB
                        RouteDurationWithService = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][1]
                                                    + RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][1]
                                                    - SortedSavingsList[entry][1])
                        #VolumetricWeight = VolumetricWeightRouteA + VolumetricWeightRouteB
                        VolumetricWeight = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][2] +
                                            RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][2])
                        #Check if restrictions are violated
                        if VolumetricWeight <= VehicleCapacity and RouteDurationWithService <= DriverWorkTime:
                            #RouteDuration = DurationA + DurationB - SavingsAB
                            RouteDuration = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][0] +
                                             RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][0] -
                                             SortedSavingsList[entry][1])
                            del RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])]  # delete old route A metrics
                            del RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])]  # delete old route B metrics
                            #If A is at the beginning of the route
                            if (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][0] == SortedSavingsList[entry][0][0]):
                                #A and B both at the beginning of the route, we flip route A and do A + B
                                if (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][0] == SortedSavingsList[entry][0][1]):
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]].reverse()  # Flip route A
                                    #RouteA = RouteA + RouteB
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] +
                                                                                               Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])
                                #A at the beginning, B at the end of route
                                elif (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][-1] == SortedSavingsList[entry][0][1]):
                                    #At route B before A
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]] +
                                                                                               Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])
                            #If A is at the end of the route
                            elif (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][-1] == SortedSavingsList[entry][0][0]):
                                #A at the end, B at the beginning of route, we do A + B
                                if (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][0] == SortedSavingsList[entry][0][1]):
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] +
                                                                                               Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])
```

Figure 30 Routing Logic (1)

```python
                                #A and B both at the end of the route, we flip route B and do A + B
                                if (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][-1] == SortedSavingsList[entry][0][1]):
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]].reverse()  # Flip route B
                                    #RouteA = RouteA + RouteB
                                    Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] +
                                                                                               Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])
                            OldRoute = CustomerInRoute[SortedSavingsList[entry][0][1]]
                            #We combine routes A and B under A's route number, so all points in route B move to A
                            for point in Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]]:
                                CustomerInRoute[point] = CustomerInRoute[SortedSavingsList[entry][0][0]]
                            #We assigned the new route to route A always, so we empty the entry of route B in the Routes list
                            Routes[OldRoute] = []
                            #We update the RouteMetrics of the newly created route
                            RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])] = [RouteDuration,
                                                                                                            RouteDurationWithService,
                                                                                                            VolumetricWeight]

            #One of locations not yet in route
            elif(((SortedSavingsList[entry][0][0] in CustomerInRoute) or (SortedSavingsList[entry][0][1] in CustomerInRoute))
                 and not(((SortedSavingsList[entry][0][0] in CustomerInRoute) and (SortedSavingsList[entry][0][1] in CustomerInRoute)))):

                if SortedSavingsList[entry][0][0] in CustomerInRoute:  # Customer A is already in a route
                    #Check if customer A is at the outside of the route
                    if ((Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][0] == SortedSavingsList[entry][0][0]) or
                        (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][-1] == SortedSavingsList[entry][0][0])):
                        #RoutedurationWithService = DurationWithServiceA + 2*HubDurationB - SavingsAB + ServiceTimeB
                        RouteDurationWithService = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][1]
                                                    + 2 * HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][1]] + 13]
                                                    - SortedSavingsList[entry][1] +
                                                    float(str(ServiceTimeArray[CustomerHubList[SortedSavingsList[entry][0][1]]]).strip('[').strip(']')))
                        #VolumetricWeight = VolumetricWeightRouteA + VolumetricWeightB
                        VolumetricWeight = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][2] +
                                            float(str(VolumetricWeightArray[CustomerHubList[SortedSavingsList[entry][0][1]]]).strip('[').strip(']')))
                        #Check if restrictions are violated
                        if VolumetricWeight <= VehicleCapacity and RouteDurationWithService <= DriverWorkTime:
                            #RouteDuration = DurationA + 2 * DurationB - SavingsAB
                            RouteDuration = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])][0]
                                             + 2 * HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][1]] + 13]
                                             - SortedSavingsList[entry][1])
                            del RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])]  # delete old routes metrics
                            #Store the new route for customer B
                            CustomerInRoute[SortedSavingsList[entry][0][1]] = CustomerInRoute[SortedSavingsList[entry][0][0]]
                            #if customer A is at the beginning of the route we add B before it
                            if (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]][0] == SortedSavingsList[entry][0][0]):
                                Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = ([SortedSavingsList[entry][0][1]] +
                                                                                           Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])
                                RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])] = [RouteDuration, RouteDurationWithService, VolumetricWeight]
                            else:  # customer A is at the end of the route, we add B after it
                                Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]] +
                                                                                           [SortedSavingsList[entry][0][1]])
                                #Store route metrics in the RouteMetrics dictionary
                                RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][0]]])] = [RouteDuration, RouteDurationWithService, VolumetricWeight]

                elif SortedSavingsList[entry][0][1] in CustomerInRoute:  # Customer B is already in a route
                    #Check if customer B is at the outside of the route
                    if ((Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][0] == SortedSavingsList[entry][0][1]) or
                        (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][-1] == SortedSavingsList[entry][0][1])):
                        #RoutedurationWithService = DurationWithService B + 2*HubDurationA - SavingsAB + ServiceTimeA
                        RouteDurationWithService = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][1]
                                                    + 2 * HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][0]] + 13]
                                                    - SortedSavingsList[entry][1] +
                                                    float(str(ServiceTimeArray[CustomerHubList[SortedSavingsList[entry][0][0]]]).strip('[').strip(']')))
                        #VolumetricWeight = VolumetricWeightRouteB + VolumetricWeightA
                        VolumetricWeight = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][2] +
                                            float(str(VolumetricWeightArray[CustomerHubList[SortedSavingsList[entry][0][0]]]).strip('[').strip(']')))
                        #Check if restrictions are violated
                        if VolumetricWeight <= VehicleCapacity and RouteDurationWithService <= DriverWorkTime:
                            #RouteDuration = DurationB + 2 * DurationA - SavingsAB
                            RouteDuration = (RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])][0]
                                             + 2 * HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][0]] + 13]
                                             - SortedSavingsList[entry][1])
                            del RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])]  # delete old routes metrics
                            #Store the new route for customer A
                            CustomerInRoute[SortedSavingsList[entry][0][0]] = CustomerInRoute[SortedSavingsList[entry][0][1]]
                            #if customer B is at the beginning of the route we add A before it
                            if (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]][0] == SortedSavingsList[entry][0][1]):
                                Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]] = ([SortedSavingsList[entry][0][0]] +
                                                                                           Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])
                                RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])] = [RouteDuration, RouteDurationWithService, VolumetricWeight]
                            else:  # customer B is at the end of the route, we add A after it
                                Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]] = (Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]] +
                                                                                           [SortedSavingsList[entry][0][0]])
                                #Store route metrics in the RouteMetrics dictionary
                                RouteMetrics[tuple(Routes[CustomerInRoute[SortedSavingsList[entry][0][1]]])] = [RouteDuration, RouteDurationWithService, VolumetricWeight]
```

Figure 31 Routing Logic (2)

```python
            #Both locations not yet in a route
            elif not((SortedSavingsList[entry][0][0] in CustomerInRoute) or (SortedSavingsList[entry][0][1] in CustomerInRoute)):
                # RouteDuration = HubDurationA + HubDurationB + DurationAB
                RouteDuration = (HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][0]] + 13] +
                                 HubDurationsMatrix[hub][CustomerHubList[SortedSavingsList[entry][0][1]] + 13] +
                                 DurationsMatrix[CustomerHubList[SortedSavingsList[entry][0][0]]][CustomerHubList[SortedSavingsList[entry][0][1]]])
                #RouteDurationWithService = RouteDuration + ServiceTimeA + ServiceTimeB
                RouteDurationWithService = (RouteDuration +
                                            float(str(ServiceTimeArray[CustomerHubList[SortedSavingsList[entry][0][0]]]).strip('[').strip(']')) +
                                            float(str(ServiceTimeArray[CustomerHubList[SortedSavingsList[entry][0][1]]]).strip('[').strip(']')))
                #VolumetricWeight = VolumetricWeightA + VolumetricWeightB
                VolumetricWeight = (float(str(VolumetricWeightArray[CustomerHubList[SortedSavingsList[entry][0][0]]]).strip('[').strip(']')) +
                                    float(str(VolumetricWeightArray[CustomerHubList[SortedSavingsList[entry][0][1]]]).strip('[').strip(']')))
                #Check if restrictions are violated
                if VolumetricWeight <= VehicleCapacity and RouteDurationWithService <= DriverWorkTime:
                    #Add AB route to end of Routes list
                    Routes.append([SortedSavingsList[entry][0][0]] + [SortedSavingsList[entry][0][1]])
                    CustomerInRoute[SortedSavingsList[entry][0][0]] = len(Routes) - 1  # Store route with customer
                    CustomerInRoute[SortedSavingsList[entry][0][1]] = len(Routes) - 1
                    #Store route metrics in the RouteMetrics dictionary
                    RouteMetrics[SortedSavingsList[entry][0]] = [RouteDuration, RouteDurationWithService, VolumetricWeight]
        #The whole SortedSavingsList has been processed, unassigned points are a route Hub-Point-Hub on their own
        for customer in range(len(CustomerHubList)):  # loop over all customers of current hub
            if not (customer in CustomerInRoute):
                Routes.append([customer])
                CustomerInRoute[customer] = len(Routes) - 1  # Store route with customer
                RouteDuration = 2 * HubDurationsMatrix[hub][CustomerHubList[customer] + 13]
                RouteDurationWithService = RouteDuration + float(str(ServiceTimeArray[CustomerHubList[customer]]).strip('[').strip(']'))
                VolumetricWeight = float(str(VolumetricWeightArray[CustomerHubList[customer]]).strip('[').strip(']'))
                RouteMetrics[customer] = [RouteDuration, RouteDurationWithService, VolumetricWeight]
        RoutesAfterBinPacking = BestFitDecreasing(Routes, RouteMetrics)
        AllRoutes[hub] = RoutesAfterBinPacking[0]
        AllRouteMetrics[hub] = RoutesAfterBinPacking[1]
    return AllRoutes, AllRouteMetrics
```

Figure 33 Routing Logic (3)

```python
def BestFitDecreasing(Routes, RouteMetrics):

    DriverWorkTime = 31500
    HubServiceTime = 1200  # Time to reload at hub

    #Sort the routingmetrics dictionary on decreasing duration
    SortedRouteMetricsList = sorted(RouteMetrics.items(), key=lambda x: x[1][1], reverse=True)

    #Wright the SortedRouteMetrics list to a dictionary again
    SortedRouteMetrics = {}
    for i in range(len(SortedRouteMetricsList)):
        SortedRouteMetrics[SortedRouteMetricsList[i][0]] = [SortedRouteMetricsList[i][1][0], SortedRouteMetricsList[i][1][1], SortedRouteMetricsList[i][1][2]]

    for route in SortedRouteMetrics:  # loop over routes
        DurationToFit = SortedRouteMetrics[route][1]
        BestFitTightness = 31500  # initialize large enough
        for bins in SortedRouteMetrics:  # loop over all routes to search for a place to fit
            #check if route fits in this bin, a route cannot fit in with itself
            if DurationToFit <= (DriverWorkTime - SortedRouteMetrics[bins][1] - HubServiceTime) and route != bins:
                FitTightness = DriverWorkTime - SortedRouteMetrics[bins][1] - DurationToFit - HubServiceTime  # This bins fit tightness
                if FitTightness < BestFitTightness:
                    BestFitTightness = FitTightness  # update BestFitTightness
                    BestBin = bins
        if BestFitTightness != 31500:  # BestFitTightness changed so a fitting bin is found
            if type(route) == int:
                routex = [route]
            else:
                routex = list(route)
            if type(BestBin) == int:
                BestBinx = [BestBin]
            else:
                BestBinx = list(BestBin)
            Routes.append(routex + ['H'] + BestBinx)  # Store the two routes together
            AIndex = Routes.index(routex)
            BIndex = Routes.index(BestBinx)
            Routes[AIndex] = []
            Routes[BIndex] = []
            SortedRouteMetrics[tuple(Routes[-1])] = [SortedRouteMetrics[route][0] + SortedRouteMetrics[BestBin][0],
                                                     SortedRouteMetrics[route][1] + SortedRouteMetrics[BestBin][1] +
                                                     HubServiceTime, SortedRouteMetrics[route][2] + SortedRouteMetrics[BestBin][2]]
            RouteMetrics[tuple(Routes[-1])] = SortedRouteMetrics[tuple(Routes[-1])]
            del RouteMetrics[route]
            del RouteMetrics[BestBin]
            #Old routes are set at the driver time limit so that they are not chosen anymore, but they are kept as an entry in
            #the dicitonary so that the loop still loops over all entries
            del SortedRouteMetrics[BestBin]
            SortedRouteMetrics[route] = [0, 31500, 0]

    return Routes, RouteMetrics
```

Figure 32 BestFitDecreasing bin packing

```python
#%%Calling code


#Give the desired configuration, including 0, in the format [x, x, ...]
Configuration = [0, 2, 3, 7, 9, 12]
#First fill in the right Excel file and the Excel sheet of the desired day in the DurationsMatrix function excel import
#Also do not forget to let the hub column in your Excel file refer to the right configuration

Result = DurationsMatrix()  # Function creates HubDurationsMatrix and DurationsMatrix
RoutingResults = RoutingLogic(Configuration, VolumetricWeights(), ServiceTime(), Result[0], Result[1])
VolWeight = CalculateTotalVolumetricWeight(Configuration, RoutingResults[1], Result[1])

AllRoutes = RoutingResults[0]
AllRouteMetrics = RoutingResults[1]

#Import pandas package for Excel import
import pandas as pd
#Excel export

for hub in AllRouteMetrics:
    RouteMetricsDataFrame = pd.DataFrame.from_dict(AllRouteMetrics[hub], orient='index')
    FileString = 'RoutingModel - Metrics' + str(hub) + '.xlsx'
    RouteMetricsDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\{}'.format(FileString),
                                   sheet_name='metrics')  # Export

VolumetricWeightDataFrame = pd.DataFrame.from_dict(VolWeight, orient='index')
VolumetricWeightDataFrame.to_excel(r'C:\Users\User\Documents\6. Master Thesis\Routingmodel - ShuttleCosts.xlsx',
                                   sheet_name='metrics')  # Export
```

Figure 34 Routing model calling procedure