Seed selection in a multi-period planning with time windows

ANNELIEKE BOSCH

MASTER THESIS
INDUSTRIAL ENGINEERING AND MANAGEMENT
TRACK PRODUCTION AND LOGISTIC MANAGEMENT

SUPERVISOR ORTEC
A. (ARJEN) RIETVELD
SUPERVISORS UNIVERSITY OF TWENTE
DR. IR. M.R.K. (MARTIJN) MES
DR. IR. J.M.J. (MARCO) SCHUTTEN

ORTEC
OPTIMIZE YOUR WORLD

UNIVERSITY OF TWENTE.

Master thesis
Industrial engineering and management

Seed selection in a multi-period planning with time windows

Annelieke Bosch

s0141607

Graduation committee:

Dr. ir. M.R.K. Mes (University of Twente)

Dr. ir. J.M.J. Schutten (University of Twente)

Drs. A. Rietveld (ORTEC)

University of Twente
Drienerlolaan 5
7522 NB Enschede
Tel. +31 (0)53 489 9111

ORTEC
Houtsingel 5
2700 AB Zoetermeer
Tel. +31 (0)88 678 3265

MANAGEMENT SUMMARY

INTRODUCTION

The planners using TRP pass four steps in generating a plan. Successively, the planners manually plan some difficult customers, TRP generates an initial solution with the sequential insertion algorithm, TRP improves this solution with the improvement steps, and the planners make some manually adjustments to the generated trips. Especially this last step cost the planners too much time at this moment. However, with these manual adjustments the planners are able not only to improve the visual attractiveness of the plan, but also the costs, number of kilometers driven, and the number of driving hours. The goal of the research is to "*find the cause why the plan generated with TRP is visually less attractive than the plan after the manual adjustments of the planners and develop an improvement of the current planning algorithm used by TRP with a focus on improving the initial solution*"

CAUSE OF THE PROBLEM

In this research, we used the customer Zeeman as leading example. We investigated the cause of the problem by analyzing the differences between the plan generated with the current algorithm of TRP and the plan that is manually adjusted by the planners of Zeeman. Two important characteristics of the planning, which make it difficult to generate a clustered and feasible plan, are the time windows of the orders and the required vehicle types for delivery of orders.

We defined indicators that examine the quality of the plan and indicators that specifically judge the extent of clustering in a plan. The four indicators of the latter are:

- the number of cities that are visited by more vehicles than required,
- the average driven distance between the first and the last order in a trip,
- the average radius of the clusters, and
- the average capacity utilization of the vehicles.

We found that on all indicators, the manually adjusted plan of Zeeman scores better than the plan generated by TRP's original algorithm. We concluded that it was not possible to identify one single cause. The most plausible explanation is that the planners explore the neighborhood of the location of the order before inserting the order into a trip, where TRP does not consider this.

CURRENT ALGORITHM OF TRP

The current algorithm in TRP consists of two phases. In the first phase, TRP generates an initial solution. The initial solution is generated with the sequential insertion algorithm. This algorithm consists of four steps:

1. Select a vehicle
2. Select a seed order
3. Assign orders to the trip
4. Move the trip to a smaller vehicle

In the second phase of TRP's current algorithm, the initial solution is improved by performing several improvement steps based on local search.

DEVELOPED APPROACHES

When we analyzed the operating procedure of the sequential insertion algorithm in the provided cases, we concluded that we tackle the heart of the problem when we improve the second step: select the seed order. The most promising solution found in literature is the circle covering method of Savelsbergh (1990). We used this method as a basis for generating the clusters in the approaches we developed. We generate a cluster for each order and determine the radius of this cluster. Next, we defined two different methods to determine which seed order to use (sequential approach) or which cluster to merge (parallel approach). In the first method, we select the cluster with the smallest radius. In the second method, we select the cluster with the largest difference between the radii of different shifts. We call the selection criterion of the method the incentive.

We developed two approaches. Figure 1 gives an overview of the approaches.

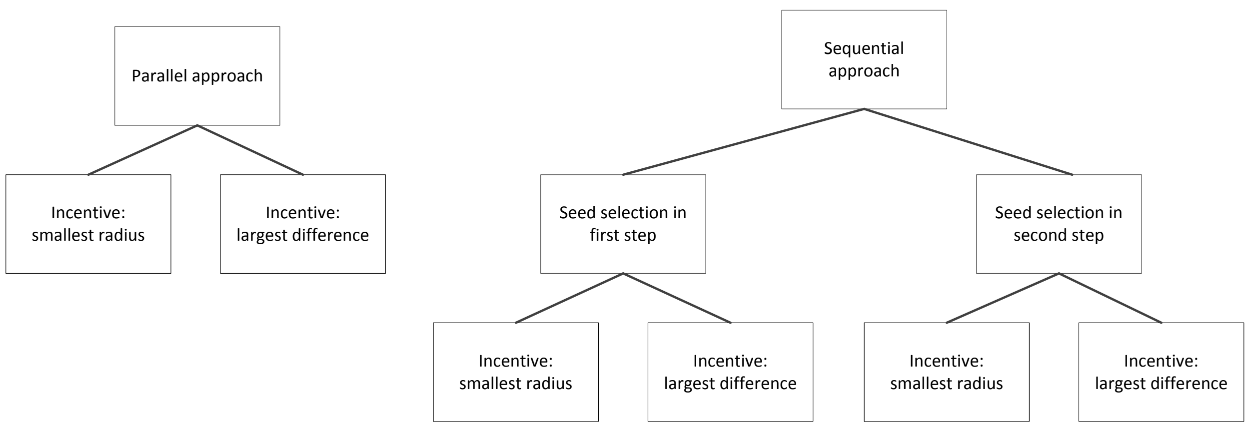


FIGURE 1 - OVERVIEW OF THE DIFFERENT APPROACHES AND METHODS

The first approach is a parallel approach. In the parallel approach, we simultaneously merge the two clusters with the highest incentive. We tested the approach with both the smallest radius and the largest difference incentive. The parallel approach shows multiple strong points on which the plan is improved. With both incentives, the visual attractiveness scores high; the solution looks more clustered. This is confirmed by the performance indicators. There are some trips with violations.

The second approach is an adjustment to the sequential insertion algorithm. We developed two variants. In the first variant, we only change the seed selection step. We use the smallest radius of a cluster or the largest difference between the radii of different shifts as selection criterion. The variant with the smallest radius does not give a feasible solution; there are too many unplanned orders because the required vehicle was no longer available. With the largest difference incentive, we overcome this problem. We again used one of the incentives as selection criterion for the seed, but in the approach, this is the first step of the algorithm and we select the vehicle in the second step.

Conclusion

The parallel approach scores relatively high on clustering, but the costs are relatively high in comparison to the sequential approach. This is mainly caused by the additional number of vehicles the parallel approach needs to plan all orders. The largest difference incentive gave for both approaches a better result. In that method, we consider both the time windows of the orders and the vehicle preference of neighboring orders in our selection process for a seed order.

We concluded that we succeed in improving the plan of TRP for the case of Zeeman. The two most promising approaches are the parallel approach with the smallest radius incentive and the sequential insertion algorithm with seed selection as first step. The sequential approach gives a better overall result, where the parallel approach gives better results with respect to clustering. It depends on the preference of the planners which method they prefer. Most planners will prefer to improve the visual attractiveness of the plan, which would plead for the sequential insertion algorithm with seed selection as first step and the largest difference incentive.

Preface

This master's thesis is the final project of my degree in Industrial Engineering and Management at the University of Twente. During my studies, I developed an interest in transportation. So, when I needed to find a company to write my master thesis, the choice for ORTEC, one of the largest providers of advanced planning and optimization software solution and consulting services in transportation and distribution, was easy. I want to use this opportunity to thank a number of people for their support during the writing of my thesis.

First, I want to thank ORTEC. Not only for the opportunity to do this research, but also for an environment which allowed me to develop myself. I would like to thank my colleagues for their support. They were always willing to help and provided me with a fun atmosphere to write this thesis. Furthermore, I would like to thank the planners at Zeeman, for giving me insight in their planning, providing me with the data for this research.

Special thanks go to my supervisors. Arjen Rietveld, my supervisor at OREC, I thank you for your support. You were always willing to make time to discuss my thesis. I also want to thank my supervisors at the University of Twente, Martijn Mes and Marco Schutten. Their constructive feedback helped me to develop a critical view on my research and improve both the content and the structure of this thesis.

Annelieke Bosch
Enschede, February 2014

Table of Contents

1. INTRODUCTION

Yearly, more than 500 billion kilograms of goods are transported over the Dutch road network (CBS, 2012). This makes road transport the most used way of transporting goods. All these goods have to be on the right place at the right time in the right amount, and transported under the right conditions. In the Netherlands, there are more than 12,000 transportation companies who take care of this (Rijksoverheid, 2011). To fulfill all those requirements of the pickup and delivery of the goods, the transportation companies need an adequate planning of their resources.

Planning is a complex task. For a long time, planners created the plan manually with the help of a map on the wall. The planning was mainly based on vision and experience of the planner. Nowadays, planning is more complex. Not only more orders need to be processed; also there are more legislations to comply with, such as regulations governing driving hours and load restrictions. With this development, a shift in priorities was set. No longer, the planning is solely based on the planners experience and the visual attractiveness of the solution, but the focus shifts more to minimization of costs. This sometimes conflicts with the experience of the planners, who say there are more aspects that should be considered than only costs. The discrepancy between the visual attractiveness and the costs makes it difficult to make a planning that satisfies the company and the planners.

This research is conducted at ORTEC. ORTEC is one of the largest providers of advanced planning and optimization software solutions and consulting services. One of its solutions focuses on vehicle routing and dispatch. ORTEC supports the planning department of various companies with its software and in that way finds a suitable solution for the transportation or distribution.

This chapter introduces the subject of this thesis. Section 1.1 briefly describes ORTEC and its transportation and distribution planning software. In the remaining of this chapter, we successively describe the motivation of the research (Section 1.2), the problem description (Section 1.3), and the problem statement (Section 1.4). We conclude this chapter with the structure of this report in Section 1.5.

1.1. CONTEXT DESCRIPTION

We conduct this research at ORTEC. The core activities of ORTEC are developing and implementing advanced planning software solutions for vehicle routing and dispatching, pallet and vehicle loading, workforce scheduling, delivery forecasting logistics network planning, and warehouse control. ORTEC provides best-of-breed, custom made, SAP® certified, and embedded solutions, supported by strategic partnerships (ORTEC BV, 2012). The solutions of ORTEC are implemented in more than forty countries all over the world. The mission of ORTEC is to "support companies and public institutions in their strategic and operational decision making through the delivery of sophisticated planning and optimization software solutions, professional consulting and mathematical modeling services" (ORTEC BV, 2012).

The research is done at the Algorithm Knowledge Team (AKT) of the ORTEC projects consulting (LPC) department. AKT is responsible for implementation and issues concerning automatic planning for all vehicle routing and dispatch products. In automatic planning, the trips in a plan are fully generated by the software of ORTEC without intervention of planner.

1.1.1. ORTEC PRODUCTS

ORTEC's solutions for vehicle routing and dispatch are bundled in three products (ORTEC BV, 2012):

- ORTEC Tactical Route Planning (TRP)
- ORTEC Shortrec
- ORTEC Transportation and Distribution (OTD)

The main difference between these three products is the planning level they focus on. Figure 2 summarizes these differences. In planning, four levels can be distinguished:

- On the strategic level, choices are made, such as where should the distribution centers be located and which product groups are assigned to a distribution center.
- On the tactical level, the planners plan all forecasted orders for a predefined period. For example, a company delivers the same customers every week. For the period of three months, the forecasted order amounts of these customers remain equal. Therefore, the plan made for the first week can be used in all following weeks during the next three months. In this plan, we consider the requirements and wishes of the customers. Based on this planning the company can make decisions about, for example, fleet optimization.
- On the operational level, a planning is made based on the real orders. The tactical planning is used as point of departure for the planning on the operational level; with other words, the tactical planning is a template for the operational planning. The adjustments in orders with respect to the tactical planning, such as changes in the order amount, sickness of drivers, or additional (emergency) orders, are considered on this level.
- On the real time level, the last-minute changes are made to the operational planning. An example is changing the route when a truck gets in a traffic jam.

FIGURE 2 - POSITIONING OF PRODUCTS (ORTEC BV, 2012)

Each company passes through all four levels of the planning. Albeit, not all companies are aware of the different levels. In some cases, the planners combine multiple levels. Figure 2 shows that OTD can be used for both the operational level and the real-time level. It is hard to draw a line between those two levels. Most customers of ORTEC have one of the products and take the decisions on the other levels in other ways, for example by using Excel, their traffic management systems, or pen and paper. Other customers of ORTEC use both TRP and OTD or Shortrec and OTD.

In this research, we focus on the planning on the tactical level. The relevant product is TRP. TRP makes relatively quickly an efficient trip schedule in which most characteristics and restrictions of the company are considered. The software aims to minimize the costs and the working time. Furthermore, TRP can be used to compare different scenarios. For example, to determine the impact of new customers, seasonal patterns, or changes in the fleet.

1.1.2. The planning process

For better understanding of the problem, we introduce the general planning process of customers that use TRP. Chapter 2 gives a more elaborate description of the planning process and the application of this process. Making a plan with TRP consists of four steps (Figure 3).

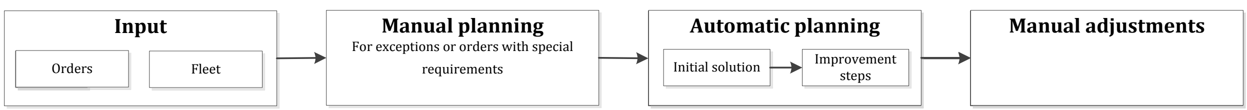


Figure 3 - Steps in the planning process

In Step 1, the planner collects the input for the algorithm and imports this in TRP. This is information about the orders, the vehicle fleet, and the customers. The order information contains the forecasted demands for the planning horizon of the tactical planning. The information about the fleet contains mainly characteristics and restrictions of the vehicles, such as the capacity of the vehicle and the region in which the vehicle can be used. Customer information are characteristics of and restrictions on the delivery address, such as delivery windows.

In Step 2, the planner makes a manual plan for some orders. This may be desirable if there are special requirements for the order. Planners have their own reasons to choose to plan an order or even a trip before the automatic planning. For example, they know that an order always provokes problems and therefore should be planned in a specific vehicle. Or the planner has preferences which are not programmed, but should be fulfilled. It is possible, but not obligated, to fixate these trips at the end of this step, such that they are not optimized in later steps.

In Step 3, we make the automatic planning. This is done in two phases: an initial plan is made and this plan is improved by multiple improvement steps. In the initial phase, a greedy solution is generated; the goal of running this phase is to get a feasible solution for the problem. However, this solution is in most cases far from optimal and solely used as a starting point for the improvement steps. In the second phase, the initial solution is improved by several improvement steps. This phase focuses on minimizing driving kilometers and working time.

In Step 4, the planner makes manual adjustments to the automatic planning generated in Step 3. This is almost always desirable. In most cases, the planners have more information and preferences that can be translated into restrictions. An example is the preference to deliver two orders by the same driver, or changing the route of a trip due to the high chance on congestion on the current route. These manual adjustments do not always improve the planning, based on driven kilometers and time, but satisfies the wishes of the planners or drivers.

The time it takes to go through this process differs per company. In most cases, generating a tactical plan takes about one day.

1.2. Research motivation

TRP is implemented at many customers of ORTEC. The majority of the planners is very satisfied with the result of the planning. However, at some customers manual adjustments allow a significant improvement. One of these customers is the store chain Zeeman.

The planners of Zeeman address that they focus on two points while they carry out the manual adjustments. First, the planners prefer to let only one truck visit a certain city, where TRP lets multiple trucks visit a city. The planners aim that a better solution is found if the number of trucks that enter a city is minimized. Second, the planners aim to minimize the number of kilometers driven between the first and the last order in a trip.

ORTEC assumes that the time windows, which most orders have, are related to the higher number of visits per city in the solution of TRP. We explain the reason with a simplified example. In the area of Enschede, Zeeman has ten orders. Five of these orders can be delivered 24 hours per day and five of these orders only between 22h and 6h. The capacity of the truck allows that all orders nearby Enschede are delivered in one trip. The algorithm used in TRP works sequentially; the first truck is filled until no order can be added, then the second truck is filled until no order can be added, and so on. When the first truck drives during day time, TRP only assigns five orders to this truck. The other five orders, which should be delivered during night time, are assigned to a second truck. The remaining capacity of both trucks is assigned to orders outside the area of Enschede. So, due to a combination of the vehicle choice and the time windows of the orders, two trucks visit Enschede.

The planners think that the capacity of the truck is not used efficiently. The planners want to minimize the number of kilometers between the first and the last order, while TRP also delivers orders on the route to the 'first' order if that is more efficient. From the planners' point of view, it is more efficient to use the capacity of the truck for orders in a specific region. A cluster denotes a grouping of orders for delivery in the same trip. In the solution of Zeeman, the density of the cluster is higher. Figure 4 and Figure 5 show a part Belgium and France in the planning of Zeeman and the planning of TRP. The red line in Figure 4 represents a trip that delivers an order in Belgium, before driving to France. The planners change the route of trip to the situation in Figure 5. The situation is visually more attractive, due to a higher density of the clusters. We discuss this situation in more detail in Chapter 4.

For both problems it holds that the planners expect that the orders are clustered in trips based on their geographical location, but that is not the case. In this research, we investigate, for the points mentioned in this section, how we need to change the algorithm in TRP such that the solution of

TRP is better attuned with the view of the planners. We investigate the exact cause of the problem in Chapter 4.

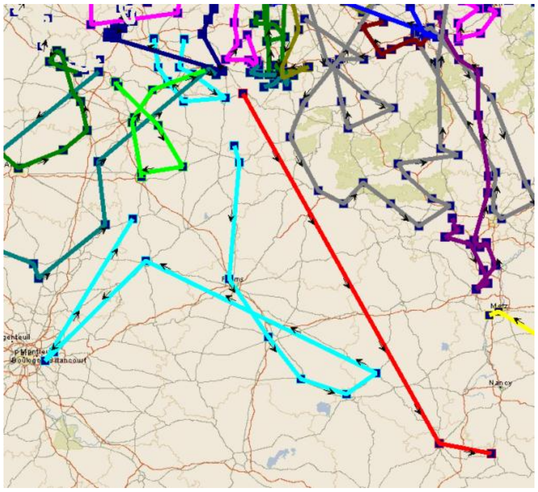


Figure 4 –A part of Belgium and France in TRP solution

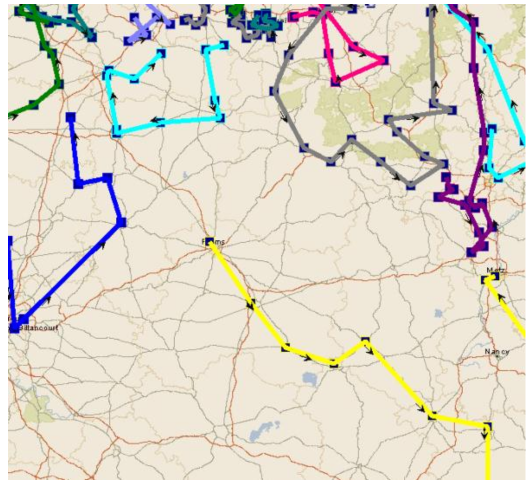


Figure 5 - A part of Belgium and France in Zeeman solution

1.3. Problem description

The planners of Zeeman improve the planning of TRP by making manual adjustments. Not only is the resulting planning more visually satisfying for the planners, also the number of kilometers driven, driving time, and the costs of the trips are decreased. For a solution promising a reduction of manual labor in the planning process, it is clearly suboptimal to produce plans which allow for significant manual improvement.

ORTEC ascribes a large part of these improvements to the reduction in the number of visits per city and the decrease of the driving distance to the next order in the planning of Zeeman compared to the planning of TRP. The problem is then to identify improvements in the planning methods of TRP with the potential to improve these two metrics.

The overwhelming majority of research effort at ORTEC to improve TRP's algorithm focuses on the improvement steps of the planning process. However, ORTEC feels that to decrease the number of visits per city and improve the capacity utilization, it is most promising to focus on the initial solution instead. Baker & Schaffer (1986) show that the quality of the final solution depends on the quality of the initial solution; problems with the best initial solution give the best overall result. This supports ORTEC's new research direction employed in this work. Consequently, the problem considered is reduced to the negative impact of the current initial solution with respect to the two metrics.

An indicator of the quality of the initial solution with respect to the metrics considered is the occurrence of clusters. Returning to Figure 4 and Figure 5, it becomes apparent why clustering can be a quality indicator. ORTEC feels that the current initial solution lacks clustering, with a likely significant impact on the results as seen by Zeeman. This brings us to our final refinement of the problem, resulting in a problem statement. This is the topic of Section 1.4.

1.4. Problem statement

In Section 1.3, we addressed that the planners improve the planning made by TRP. The planners focus on making a more clustered solution. This research explores this and develops an improvement of the current algorithm to tackle the problem. The objective of this research is:

"*Find the cause why the plan generated with TRP is visually less attractive than the plan after the manual adjustments of the planners and develop an improvement of the current planning algorithm used by TRP with a focus on improving the initial solution*"

To achieve this objective we formulate five research questions.

1. **What is the current planning process at Zeeman?**
   This question aims to get an overview of the planning process in the current situation. We describe the planning process of the main case for this thesis, Zeeman. To retrieve this information, we visit Zeeman and interview the planners. Subsequently, we describe in detail the way of working of the automatic planning algorithm in TRP.

2. **Why is the visually attractiveness lower in the current planning of TRP?**
   This question aims to get more insight in the cause of the problem that the plan of TRP is visually less attractive. We use different datasets of Zeeman to find the cause. We compare the manually adjusted plan with the plan made by TRP. We define performance indicators to compare the planning among others on the number of visits per city, the driving distance between the orders, and the capacity utilization. Furthermore, we compare the visual attractiveness of the different plans and compare trips that caught our attention during the analysis.

3. **What is known in literature about clustering in vehicle routing problems with time windows?**
   This question summarizes the current state of the art in academic literature with respect to clustering in vehicle routing problems with time windows. The aim is to introduce the reader into the subject of clustering in vehicle routing problems with time windows and to identify a direction for our solution approach. During the literature search, we keep the characteristics from the planning (find by answering Question 2) in mind.

   We describe the adjustments of the sequential insertion algorithm which we found in literature. Furthermore, we describe some alternative algorithms to generate an initial solution which we found in literature.

4. **How should an approach for the clustering of orders in a vehicle routing problem with time windows look like?**
   This question develops approaches to improve the clustering of orders in a vehicle routing problem with time windows. We base our approaches on the information we found by answering the first three research questions.

5. **How do the developed approaches perform on the used datasets?**
   This question aims to get insight in the quality of the developed approaches. We apply the approaches developed by answering Question 4 on a dataset of Zeeman. We compare the planning with the current plan made by TRP. We use the indicators and comparisons as we used by answering Question 3.

Each research question is a separate chapter in this thesis. The flow of the research is as indicated as in Figure 6. The numbers indicate the different research questions, while the arrows represent the approach taken to answering them. We iteratively develop a solution approach, validate this approach and improve the solution approach based on our findings. The reader is not expected to trace this loop in the thesis. The flow for the reader is outlined in the next section. During this research, we use the case of Zeeman as leading example.

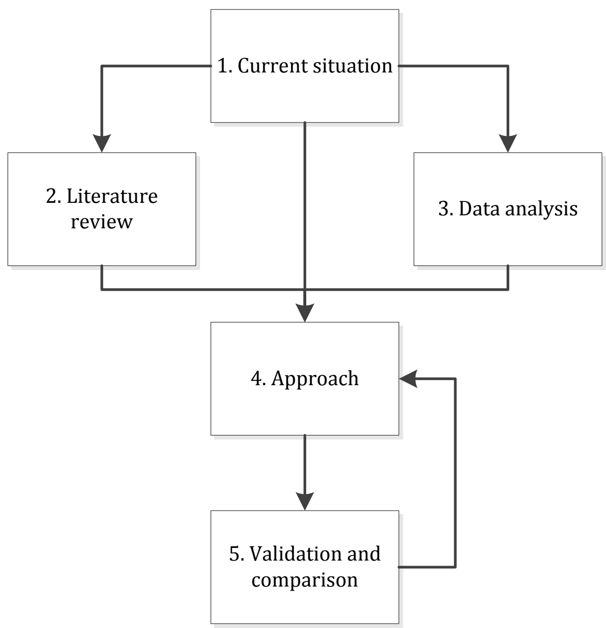


FIGURE 6 - RESEARCH FLOW

1.5. OUTLINE OF THE THESIS

The remainder of this thesis is structured as follows. Chapter 2 describes the current planning process at Zeeman and the algorithm that TRP currently uses. Chapter 3 describes the relevant literature for this thesis. This chapter is followed by a data analysis in Chapter 4. In the data analysis, we find the cause of the problems. Subsequently, we describe improvements for the approach in Chapter 5. Chapter 6 validates and compares the developed approaches. We conclude this thesis with a conclusion and some recommendations in Chapter 7.

2. The current planning process of Zeeman and TRP's algorithms

In this thesis, we aim to find a general solution for all customers of ORTEC that have a multi period planning with time windows and want a clustered solution. Zeeman is a good example of such a customer. For better understanding of the algorithm used in TRP, we describe the planning process at Zeeman in this chapter. Section 2.1 describes the input of the planning process of Zeeman. Section 2.2 describes the planning process. In Section 2.3, we explain the algorithm used for automatic planning in TRP. We end this chapter with the conclusion in Section 2.4.

2.1. The input for the planning process

Zeeman is a retailer originated in the Netherlands. In 1967, Zeeman opened the first store in Alphen aan den Rijn. Nowadays, they have over 1200 stores in the Netherlands, Belgium, Luxembourg, Germany, and France. The depot of Zeeman is still located in Alphen aan den Rijn. To supply all these stores, Zeeman generates its planning following a specific method. Three types of inputs are necessary to generate a planning: the orders, the vehicles, and the stores.

2.1.1. Orders

Zeeman has a multi-period planning. All stores of Zeeman are supplied twice a week, except the stores in the south of France. Each supply is called an order. The day of the week the orders are supplied is not determined on forehand as long as it is possible with the time windows. Most orders have time windows in which they should be delivered. The time windows contain information about the day of the week, or the time of the day it is possible to deliver an order. Some orders only have limitations on one of those two aspects, others on both. Each time window consist of a weekday and a time interval. For example, an order that needs to be delivered on a Tuesday has a time interval from Tuesday 0:00 to Tuesday 23:59. An order which needs to be delivered between 10:00 and 22:00, but with no restriction on the weekday, has seven time windows: Monday from 10:00 to 22:00, Tuesday from 10:00 to 22:00, and so on. The time windows of the orders are hard, which means that it is not allowed to start the delivery of an order earlier than the beginning of the time window, or finish the delivery of an order later than the end of the time window. About half of the orders at Zeeman have a specific time of the day in which they should be delivered. For the orders without time windows it does not matter on which day of the week they are supplied, as long as the time between two supplies is more than 24 hours.

Each order consists of an amount of trolleys. These trolleys are used to transport the different products that need to be replenished in the stores. Different types of products can be stored in one trolley. For making a tactical planning, it is therefore not interesting to know which products are transported, only how many trolleys are filled. Zeeman distributes about 7,500 trolleys per week.

The stores of Zeeman are divided into four regions, see Figure 7. Each region is assigned to a specific vehicle type. We explain the reasoning behind this and the deviations of the vehicles in Section 2.1.2. The blue cross in Figure 7 is the depot in Alphen aan den Rijn. The first region encloses a large part of the Netherlands and Flanders in Belgium. The second region encloses the stores in the north of the Netherlands, and the remaining stores in Belgium, and the stores in Ruhr in Germany. The remaining stores in Germany and the stores in the north of France are in the third region. All remaining stores in France form the last region.

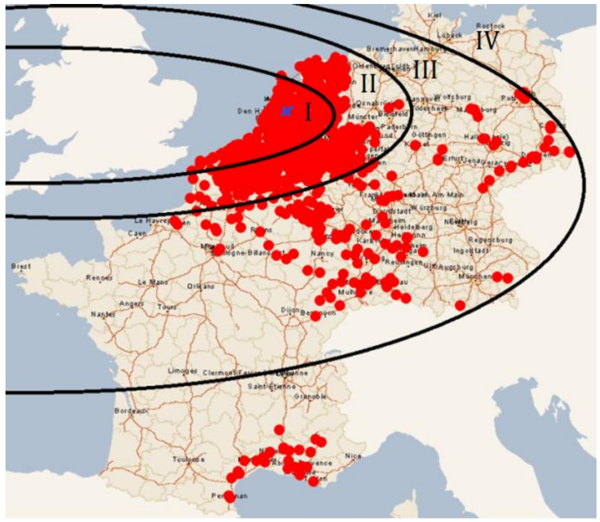


FIGURE 7 - REGIONS

2.1.2. FLEET

Zeeman has a heterogeneous fleet of 46 trucks. There are three different types of trucks: box trucks, truck and trailer combinations, and longer heavier vehicles. The main difference between those three types of trucks is the capacity.

BOX TRUCKS

The shifts of the box trucks are always during day hours. The capacity of the box truck is 36 trolleys. They deliver specific customers. For example, stores located in the city center, which are hard to reach by truck. The planners know from experience which orders to select for those vehicles. The planners manually plan these trips.

LONGER HEAVIER VEHICLES

Subcontractors drive the longer heavier vehicles. These trucks are longer than a standard truck and trailer combination and they are allowed to transport more weight. The capacity of the longer heavier vehicles varies from 46 to 90 trolleys. They deliver specific customers. For example, customers in a region with a low density of Zeeman stores. Just like the box trucks, the planners know from experience which stores to select and plan these stores manually in those vehicles. Among others, the orders in Region IV are delivered by subcontractors.

TRUCK AND TRAILER COMBINATION

TRP plans the trips for the truck and trailer combinations. The capacity of a truck and trailer combination is 46 trolleys. The truck and trailer combinations drive three different types of shifts. The first possibility is called shift work. The duration of this shift is shorter than twelve hours. The driver only has a short lunch break. The orders in Region I are delivered by vehicles that drive this shift. A truck trailer combination can also be deployed for – what Zeeman calls – a short multiple day shift. These shifts have in most cases a duration of about 24 hours. During this shift, the driver has a long rest break of nine hours, so he can have some sleep and two shorter breaks. The orders in Region II are delivered by vehicles with a short multiple day shift. The last type of shift is called a long multiple day shift. The vehicles assigned to this shift drive to the stores farther away and they stay away for three days. The driver has about two long rest breaks in which he can have some sleep and five shorter breaks. The orders in Region III are delivered by vehicles that drive this shift. Most long multiple day vehicles are manually planned by the planners.

Although, the three shifts mentioned above are deployed by vehicles with the same characteristics, we treat these shifts in the remaining of the research as like they are different vehicles. This is similar to the way Zeeman handles this situation. In total, Zeeman has twelve truck and trailer combinations available to execute a shift work shift. Since there are twelve different shift work shifts, we say we have 144 shift work vehicles. There are also twelve truck and trailer combinations available to drive a short multiple day shift. With four different shifts available, this results in 48 vehicles. Finally, we have twelve truck and trailer combinations available for long multiple day shifts, of which we have three in total. This results in 36 long multiple day vehicles. The total input of the planning of Zeeman consists of 228 vehicles.

Each vehicle has an identification number. For example, the vehicle “short multiple days 375” is a truck and trailer combination, used for a shift work shift from Tuesday 10:00 to Wednesday 10:00. In this example, 375 is the identification number of the vehicle. The numbering of the identification code is chronological.

Furthermore, each vehicle type has a priority. The long multiple days vehicles have the highest priority (5), followed by the long multiple day (4), short multiple day (3), shift work (2), and finally the box truck (1). The algorithm needs this priority to ensure that an order in a region is delivered by the desired shift type. Table 1 summarizes the information of Section 2.1.

2.2. The planning process

In Section 1.1.2, we introduced the flow of the different stages of the planning process. In this section, we discuss the planning process for the tactical planning in more detail. Figure 8 represents this process in a flowchart. We use the input we defined in Section 2.1.

First, the planners of Zeeman manually plan all box trucks, subcontractors, and some long multiple day vehicles. Successively, the automatic planning of TRP makes a plan for the first half of the week. The planners may make manual adjustments to this plan if they think this is necessary. In this plan, each store is supplied once. Since most stores are supplied twice a week, we copy the planning of the first half of the week for the second half of the week; the same trips can be made. If necessary, some additional manual adjustments are made by the planners. When the planners are satisfied with the planning, it is saved and this template of trips is sent to OTD. This whole process takes about four days. About three of these four days, the planners make manual adjustments to the planning. In OTD, the actual planning per day is made. If adjustments are made in the planning in OTD this is mainly caused by a difference between the expected and the realized order amounts or

Table 1 - Vehicle information

Priority	Shift name	Vehicle type	Region	Frequency of supply
1	Box truck	Box truck	Selection of stores	Twice per week
2	Shift work	Truck trailer combination	Large part of the Netherlands and parts of Belgium (Flanders)	Twice per week
3	Short multiple day	Truck trailer combination	North of the Netherlands, Belgium and Ruhr	Twice per week
4	Long multiple day	Truck trailer combination	Germany and north of France	Twice per week
5	Subcontractors	Long heavier vehicles	South of France	Once per week

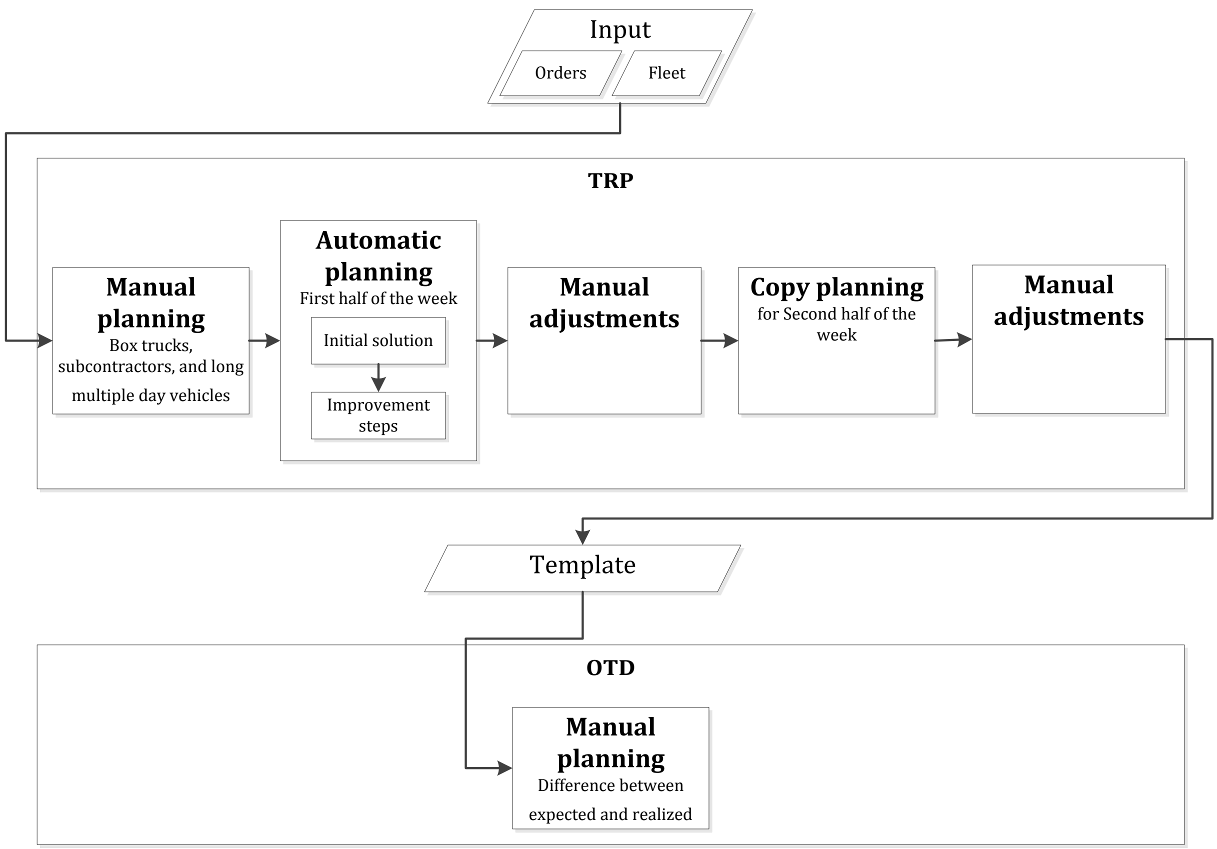


Figure 8 - Planning process of Zeeman

sickness of a driver. The focus of this research is on generating a planning on the first half of the week, done by the automatic planning of TRP. In the remainder of this thesis, we use the term TRP for the automatic planning module of the software.

2.3. THE PLANNING ALGORITHM OF TRP

A large part of the planning of Zeeman is generated with TRP. As introduced in Chapter 1, the planning algorithm of TRP consists of two phases: generate an initial solution and improve this solution. The goal of the initial solution is to make a feasible plan. Although a better initial solution results in general in a better plan, the quality of the plan is less important. There are two algorithms programmed in TRP to generate an initial solution. Zeeman uses the sequential insertion algorithm (Section 2.3.1). The other algorithm is the savings algorithm (Section 2.3.2). The choice of one of these algorithms depends on the characteristics of the planning. We contrast these two algorithms in Chapter 3. The initial solution is improved in several steps (Section 2.3.3).

2.3.1. THE SEQUENTIAL INSERTION ALGORITHM

The sequential insertion algorithm is an algorithm that is often used in vehicle routing problems. The algorithm consists of four steps which are iteratively performed to generate all trips in the plan. Figure 10 depicts these steps. We describe the interpretation of ORTEC of these four steps below (Poot, Kant, & Wagelmans, 2002):

1. Select a vehicle based on following the criteria (a is most important):
    a. Select the vehicle with the highest vehicle priority
    b. Select the vehicle with the largest available capacity
    c. Select the vehicle with the largest capacity
    d. Select the vehicle with the lowest identification number

   If we do not found a vehicle to which we can assign orders, we quit the algorithm;
2. Select a first order for this vehicle. We call this order the seed order. For this, the most difficult order should be found. In TRP, the most difficult order is the order farthest away from the depot. This seed order should satisfy all four statements below.
    a. The order is not assigned to a vehicle yet
    b. The order is farthest away of the depot among the non-assigned orders
    c. The order can be delivered during the shift time of the vehicle
    d. The order amount is lower than the remaining capacity of the truck

   If no seed order is found, go back to Step 1;
3. Assign the seed order (selected in Step 2) to the vehicle (selected in Step 1), and add orders to the new trip. First, a set of orders that is feasible to insert in the current trip is defined. Only orders that satisfy the following conditions are added to the list:
    a. The order can be delivered during the shift of the vehicle
    b. Insertion of the order will not lead to exceeding the capacity of the vehicle

   For all these orders the insertion costs are calculated with Equation 1, in which i is the order before the insertion and j the order after. l is the order that will be inserted between order i and j (Figure 9). Finding the order with the cheapest insertion costs in TRP differs from the most well-known way to find an order to insert. TRP takes the location of the seed order as point of departure and search for the unplanned order that is closest to the seed order and keeps the trip feasible after inserting the order. Subsequently, TRP calculates the insertion costs for inserting that order in all points of the route. The order is added to the point of the trip with the lowest insertion costs. Further, all orders are sequentially added to the trip.

$$insertion\ costs = c_{il} + c_{jl} - c_{ij} \quad \textbf{Equation 1}$$

Figure 9 – Locations

We repeat Step 3 until the vehicle capacity is fully used, or when adding an additional order leads to restriction violations, or when we cannot find a feasible order after a fixed number of tries (1000 in the case of Zeeman).

4. Move the trip to the smallest feasible vehicle. In this, the priority of the vehicle is no longer an issue. If we can move the orders to a smaller vehicle, we go back to Step 3 to find if we can add additional orders to trip. Otherwise, we go back to Step 1.

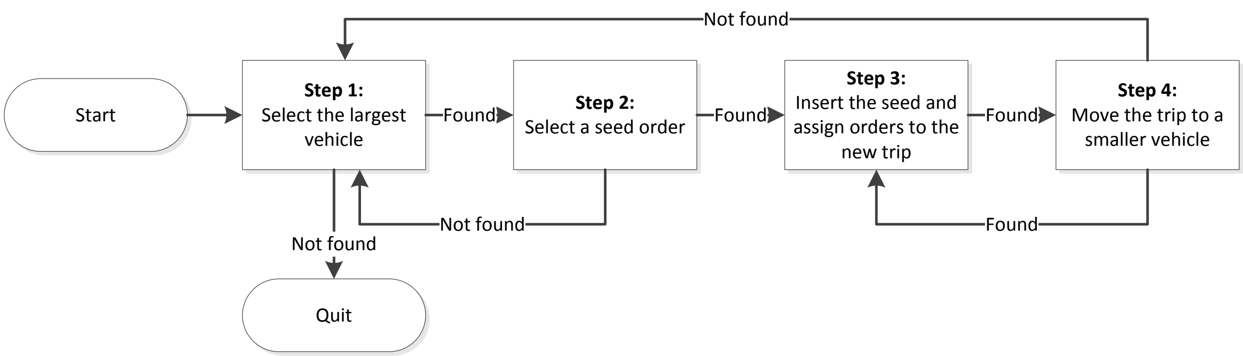


Figure 10 - The four steps of the Sequential Insertion Algorithm (ORTEC BV, 2011)

In most cases, some customer specific restrictions or selection criteria are added to the steps above to reduce the calculation time. The sequential insertion algorithm is relatively fast and can be easily understood and easily be implemented (ORTEC BV, 2011).

2.3.2. The savings algorithm

Next to the sequential insertion algorithm, there is another algorithm programmed in TRP, the savings algorithm. This algorithm is based on the savings method by Clark and Wright (1964). Figure 11 gives the eight steps of the savings algorithm.

The savings algorithm starts with a solution in which each order is assigned to a separate route. Next, we calculate the savings from combining trips. The savings from combining trip *i* and trip *j* is defined as $savings\ S_{ij} = d_{i0} + d_{0j} - d_{ij}$. d_{i0} is the distance from the last order in the trip to the depot, d_{0j} the distance from the depot to the first order in trip *j*, and d_{ij} the distance from the last order in trip *i* and the first order in trip *j*. Subsequently, we select the trip with the highest saving and assign these to the best available feasible vehicle. The information is updated and we repeat the process until no feasible combination of trips is found any longer.

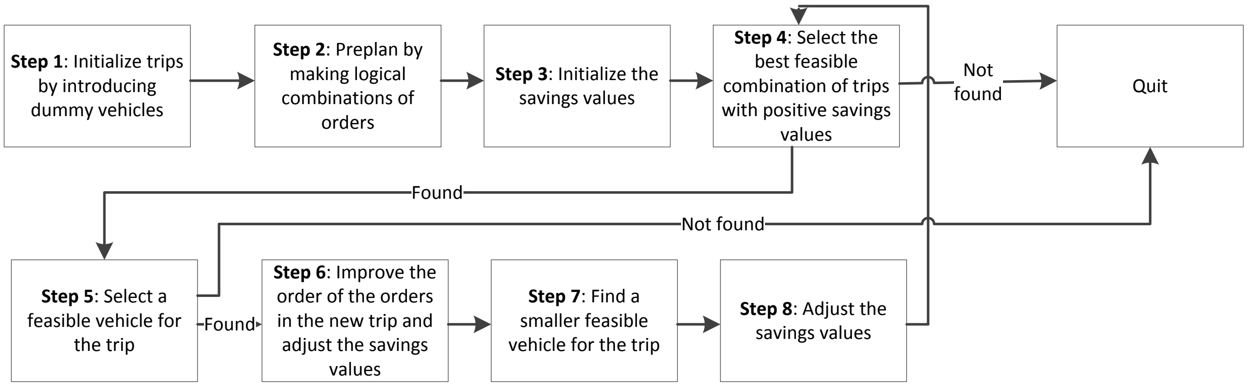


FIGURE 11 - THE EIGHT STEPS OF THE SAVINGS ALGORITHM (ORTEC BV, 2011)

2.3.3. IMPROVEMENT STEPS

The initial solution is a feasible solution that is used as point of departure. In TRP, it is possible to define the sequence in which the improvement steps are executed. It is possible to execute certain steps multiple times or to skip steps. Per customer this sequence is fixed.

The objective of the improvement steps is to reduce costs. After every step, the current solution is evaluated. The new solution is only accepted when the costs of this solution are decreased compared to the previous solution. If that is the case, the improved solution is the starting point of the next step. If that is not the case, the previous solution is used. A disadvantage of this approach is that a step that not directly leads to a better solution, but may give a better solution after and additional improvement step is performed, is ignored.

Before we describe the improvement steps, we first need to introduce some terminology. We use an example to clarify the terms. In the example, we have an initial solution with two trips, Trip A (green) and Trip B (orange). In each trip, we deliver four orders. When we exchange orders, we trade an order in Trip A for an order in Trip B (Figure 12). The red order in Trip A is switched with the red order in Trip B. The total number of orders in each trip remains four. When we move orders, we remove an order in one trip and insert this order in another trip. In our example, we moved the red order from Trip A to Trip B (Figure 13). Now, Trip A consists of three orders and Trip B consists of five orders.

In the improvement phase at Zeeman, the steps described below are performed. In Appendix A, we depict the actions performed when executing a step.

1. Optimization within a trip: we change the position of the orders in a trip
2. Move orders between trips: one order is moved to another trip
3. Exchange vehicles: exchange the vehicles of two trips

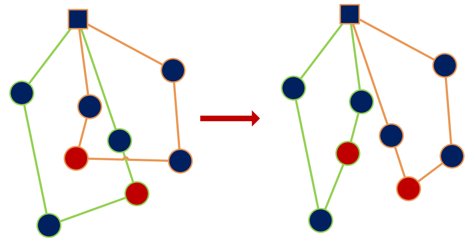


FIGURE 12 - EXCHANGE ORDERS

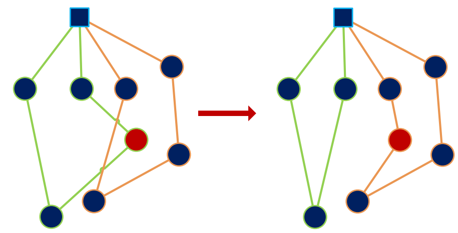


FIGURE 13 - MOVE ORDERS

4. Exchange orders: trade a number of orders in one trip with the same number of orders in another trip
5. Optimization between trips: this is a combination of Step 2 and 3. First exchange the vehicles of two trips and subsequently exchange orders between those two trips
6. Choose cheapest vehicle for a trip: move the trip to an empty vehicle
7. Flip trips: the order within the trips is reversed

These steps are executed in the following sequence: SR- 1-2-3-1-4-5-6-4-7-2-ER-4-2-3-1-4-1. In this, SR stands for start recurrence and ER for end recurrence. The steps between SR and ER are repeated five times.

2.4. CONCLUSION

In this chapter, we answered the question: "*What does the current planning process at Zeeman looks like?*". We described the orders and the fleet of Zeeman. This is the input for the tactical planning made in TRP. The planners manually plan the box trucks, the subcontractors and some long multiple day vehicles. The focus of this research is on the automatic planning, and thus on the planning of the shift work and short multiple days vehicles. The planning process is summarized in Figure 8. The automatic planning is generated in two steps. First, an initial solution is made with the sequential insertion algorithm. Subsequently, this solution is improved in multiple local search steps.

3. LITERATURE REVIEW

In Chapter 2, we described the planning process and TRP's algorithms. In this research, we focus on improving the initial solution of TRP, which is generated with the sequential insertion algorithm. We have two options to gain a better initial solution. Either we replace the algorithm that generates the initial solution by a completely new algorithm, or we adjust a step of the algorithm we currently use. In this chapter, we discuss the literature relevant for vehicle routing with time windows with the focus on clustering.

In Section 3.1, we summarize the most important characteristics of the planning. These characteristics are our starting point in the search for relevant articles. Section 3.2 contrast the sequential approach and the savings algorithm, which are both programmed in TRP. We discuss the adjustments that are made to the sequential insertion algorithm in literature in Section 3.3. Successively, we introduce some alternative algorithms which focus on clustering (Section 3.4) and we discuss literature concerning seed selection (Section 3.5). The conclusions of this chapter are in Section 3.6.

3.1. CHARACTERISTICS OF THE PLANNING

The main goal of this research is to make a better planning in TRP, such that the planners need to make less manual adjustments. The most important characteristics of the planning of the cases we consider are:

- Multi-period planning
- Orders are delivered in different shifts
- Vehicle fleet is fixed and heterogeneous
- Specific regions might require delivery with a specific vehicle type
- Single depot
- Orders are scattered over multiple regions, the density of the orders differs per region
- Single capacity constraint
- Orders may have time windows in which the delivery should take place

3.2. DIFFERENCES BETWEEN THE TWO ALGORITHMS PROGRAMMED IN TRP

In Chapter 2, we introduced the two algorithms that are programmed in TRP: the sequential insertion algorithm and the savings algorithm. In literature, different arguments are named in favor and disfavor of these two algorithms. The sequential insertion algorithm is commended because of its simplicity and ease of implementation (ORTEC BV, 2011). Furthermore, the ratio of the solution quality and the calculation time are high. A minus point is that the trips can be visually unattractive, for example due to trip crossings. On the other hand, the savings algorithm scores relative good on this point. Solomon (1987) assigns these differences to the way the orders are assigned to the routes. The insertion algorithm constructs the routes sequentially, while the savings algorithm constructs the routes parallel. The sequential insertion algorithm selects a seed order and subsequently selects the orders that fit best in that vehicle. The savings algorithm searches in each step for the best order to insert and subsequently search for the best route is to insert this order in.

A disadvantage of the savings algorithm is that the assignment criterion is only based on distance or only based on driving time, never on both. It may be that orders which are nearby in kilometers are far away according to time. This may lead to long waiting times in plans with orders with time windows. Different solutions are carried out to minimize the waiting times. Solomon (1987) compared the savings algorithm with the algorithm with restrictions on waiting time. With that adjustment, the score on waiting time is slightly better than with the insertion algorithm. However, on other points the savings algorithms did not perform well in combination with time windows. The most important point is that the savings algorithm with time windows requires more vehicles in almost all cases.

3.3. ADJUSTMENTS TO THE SEQUENTIAL INSERTION ALGORITHM

Multiple authors suggest improvements for the sequential insertion algorithm. Most of them developed smarter ways to select the orders that are inserted into the routes. Iaonnou, Krikitos, and Prastacos (2001) select the order that minimizes the impact, instead of simply inserting the order with the lowest cost to the emerging route. In this selection process, the time windows of the orders play an important role. The impact is defined in two criteria. The first criterion identifies the best order to be inserted in the current route by measuring the coverage of the selected order's time window. The second criterion determines the best insertion place in the current route. For the latter, again two criteria are relevant: the average length of the unutilized time window over all non-routed orders and a weighted combination of the additional distance, the marginal time feasibility, and the time window compatibility.

Potvin and Rousseau (1993) extended the sequential insertion algorithm, by making a parallel variant of the algorithm. Potvin and Rousseau use the solution of the sequential insertion algorithm as initial solution. This solution gives an upper bound for the number of routes. The farthest order in each route is determined. These orders are used as seed orders. Subsequently, for each unplanned order, its best feasible insertion place is determined. The insertion criteria are almost equal to those of the sequential variant. However, a generalized regret measure is added. This factor sums the difference between the best alternative and all other alternatives. The order with the largest difference is inserted. According to Potvin and Rousseau, the sequential approach works better in the situation where the orders are clustered. When that is not the case, the parallel approach works better.

3.4. CLUSTER ALGORITHMS

In Section 2.1, we introduced the two algorithms that are implemented in TRP. When we explore literature, we find a lot of alternatives. Roughly, they can be divided into two categories: traditional heuristics and metaheuristics. The traditional heuristics are the more simple heuristics, such as the sequential insertion heuristic. They provide good solutions with a low computational effort (Bräysy & Gendreau, Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms, 2005). Most metaheuristics are based on a traditional heuristic. In general, the quality of the solution of the metaheuristics is higher. However, more computational effort and time is needed to generate this solution. The focus of this research is on creating a better initial solution for the planning of TRP. This solution should be feasible, but since we improve the initial solution with the improvement steps, the initial solution does not have to be the (near) optimal solution.

Therefore, the focus in the chapter is more on the traditional heuristics. However, research of Baker & Schaffer (1986) shows that the quality of the solution depends on the quality of the initial solution; problems with the best initial solution give the best overall result. We should make a consideration between the simplicity of the algorithm of the initial solution and the quality.

3.4.1. Other route construction algorithms

Traditional heuristics are in the classes of the savings heuristics, nearest-neighbor heuristics, insertion heuristics, and sweep heuristics. In the course of time, a lot of adjustments are made to these heuristics. An important adjustment for this thesis is the addition to consider the time windows of the orders. Solomon (1984) is one of the first authors who made adjustments to these heuristics in that direction. We already introduced the insertion heuristic and savings algorithm in Chapter 2. The nearest-neighbor heuristic uses the order closest to the depot as seed order. The order closest to the seed order is added to the route. Subsequently, the order closest to the order last inserted to the route is added at the end of the emerging route. This is repeated until the capacity of the truck is reached or another restriction is violated. Closest may be defined in distance or time. The last class of algorithms is the sweep algorithms. These algorithms use a cluster first, route second algorithm approach. This means that successively groups of orders are created and for each group the sequence of the delivery of the orders is determined. The most well-known sweep algorithm is the one of Gillet and Miller (1974). In their heuristic, a seed order is selected based on their polar-coordinate angle. With forward or backward sweeping, they select the next order. When the capacity of the truck is reached, or another restriction is violated, the next seed order is selected. When all orders are assigned to a group, the route is optimized for each group.

Solomon performed a test to determine the performance of the traditional heuristics. According the test, the nearest neighbor heuristic performs not that well. The solution quality of this heuristic is lower than the quality of other heuristics. Not only is the computation time of the nearest neighbor heuristic relative long in comparison to the savings, insertion, or sweep heuristic, but there is also a relative high deviation from the best solution value found with the other heuristics.

In general, the sweep algorithm gives good results, although the sequential insertion heuristic scores better on solution quality and computation time. The test of Solomon (1984) shows that the sweep algorithm scores worse in scenarios with tight time windows and a short scheduling horizon. In cases with larger time windows, the results of the sweep algorithm approach are quite similar to the results of the sequential insertion heuristic. An advantage of the sweep algorithm is that the focus is more on clustering in comparison to the savings and insertion algorithms. A disadvantage of the algorithm is that Gillet and Miller only use the polar coordinates angels and therefore, they not consider the distance from the orders to the depot. This is clarified with an example. Figure 14 gives an overview of a set of orders in the south west of the Netherlands and Belgium (the red dots). In this figure, we draw the cones as they would look like when using the first step of the sweep algorithm. The cones are oblong. When creating a route within a cone, the route is also oblong. In one of the cones, a black line is drawn. This is the route of the trip in that cone. Figure 15 shows a more convenient route. Here, the orders are clustered more horizontally instead of in the oblong cones. This solution saves kilometers and time since the trucks only has to drive to this part of France once, while this would be about six times in the solution of Figure 14.

Newell and Daganza (1986) studied the optimal shape of a zone. They stated that the zones should approximately take the shape of a wedge and form a ring-radial type of partition. All these zones combine to a circle with multiple rings (Figure 16). Newell and Daganza (1986) are not clear how to determine the number of rings. They do give a formula to determine the size of each wedge. This size is based on the number of observed orders in a ring.

Since the sweep algorithm works with polar coordinates, it is hard to implement zones into the sweep heuristic. Fisher and Jaikumar (1979) developed the generalized assignment method. Just like the sweep algorithm, this method divides the orders into cones. Since they use a different process to create the cones, it is possible to uses zones. First, Fisher and Jaikumar (1979) determine the number of cones needed. This number is solely based on the capacity of the truck. This is a disadvantage of the algorithm, since other criteria as time windows are not considered. Subsequently, for each cone a seed order is selected. This seed order is always in the middle of the cone. Third, the order with the lowest insertion costs is assigned to the seed order. In the original algorithm, Fisher and Jaikumar work with only one ring. However, since they work with coordinates instead of angels, it is easier to divide the area into multiple cones. For example, one determines the rings by setting a radius from the depot. All orders located within the radius are assigned to that ring. From that moment the original steps of the algorithm of Fisher and Jaikumar can be used.

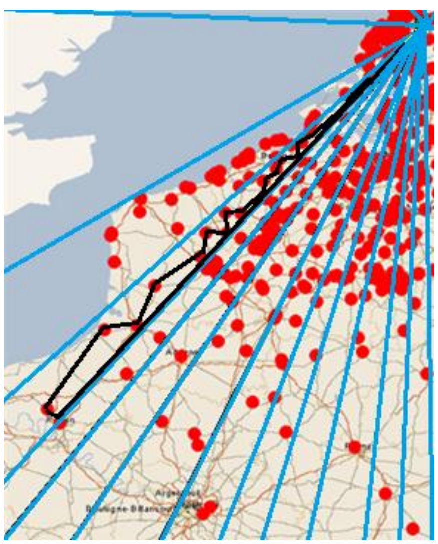


FIGURE 14 - EXAMPLE OF THE CONES IN THE SWEEP ALGORITHM

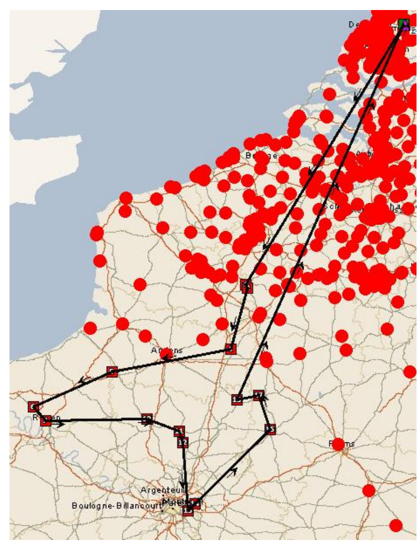


FIGURE 15 - EXAMPLE OF A ROUTE

In the paragraph above, we already mentioned that the cones in the algorithm of Fisher and Jaikumar are solely based on the capacity of truck. Multiple authors thought of a variant of the heuristic of Fisher and Jaikumar that considers time windows. Most authors develop a two phase procedure. In the first phase, most authors apply the algorithm of Fisher and Jaikumar. All orders are assigned to a cluster, but in the routing phase of the algorithm it is not possible to find a feasible solution for some orders. In the adjusted algorithms, a second step is added. In this step, all unplanned orders are assigned to a trip. Zhong and Cole (2005) develop a procedure in which new routes are inserted iteratively until a feasible solution is found. Solomon (1984) thought of a two phase procedure in which the algorithm of Fisher and Jaikumar is applied in both the first and the

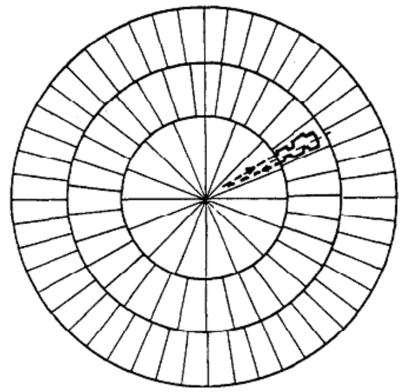


FIGURE 16 - RING-RADIAL (NEWELL & DAGANZO, 1985)

second step. In the second step, other seed orders are selected. There are also authors who consider the time windows during the clustering phase. A disadvantage of that approach is that it makes the algorithm complex since a lot of calculations have to be done to determine whether a cluster is feasible or not. In most situations this leads to a parallel approach.

Bramel and Simchi-Levi (1995) do not work with cones or zones. They developed the location based approach. In this approach, a number of zones are selected and a seed set is set for each zone. Subsequently, the capacitated clustering problem is solved. In the capacitated clustering problem, we select some of the seed orders and assign the non-seed orders to a chosen seed order, such that the costs from the depot to the seed order and from the seed to each assigned non-seed order is minimized. However, Bramel and Simchi-Levi do not describe in their article how the zones are defined. Section 3.5 discusses possible methods for defining seed order.

3.4.2. METAHEURISTICS

As we already mentioned briefly in the introduction of Section 3.4, metaheuristics in general have a better solution quality than the traditional approaches. However, the computation time needed to find this solution is longer. In literature, we find a lot of metaheuristics. In most of the metaheuristics, two or more (traditional) heuristics are combined to find a new heuristic with a better solution quality. In literature about metaheuristics that consider time windows, most heuristics successively generate an initial solution and improve this solution with improvement steps.

This section gives only a very minimal selection of all literature available about metaheuristics. It is not possible to discuss all directions of the solution approaches available in literature. The solution to the problem of this research focusses on the initial solution. Most metaheuristics are too complex or need too much computation time to consider as initial solution for this research. However, we discuss some of them below, because the way of thinking of the authors may be helpful in finding a solution to our problem.

One of the classes in metaheuristics are the local search methods. The improvement steps of TRP (Section 2.3.3) are an example of a local search method. The most common class of the local search algorithms are the edge-exchange algorithms. Here, a number of orders in the initial tour are replaced by another set of orders. In local search methods, the solution is iteratively improved by exploring neighborhood solutions. A risk when using local search methods is that the method stops at a local optimum, and consequently does not find the optimal solution. Therefore, the initial solution is important in these methods. When starting with a bad initial solution, a near optimal solution will not be found. It is proven that problems with the best initial solution give the best overall result (Baker & Schaffer, 1986).

The focus of most literature about local search in vehicle routing is on the search steps to improve the solution. Only little attention is paid to the way the initial solution is generated. Two algorithms which have a high incidence in literature are the sequential insertion algorithm and the algorithm of Fisher and Jaikumar (1981).

Another type of local search method which scores good on solution quality, is the three phase local search approach. As the name of the approach already suggests, this approach works with three phases. The first phase generates an initial solution. In the second phase, most of the initial solution is thrown away. Only some strong aspects are used as input for the second phase. In literature, we find multiple examples that all use other aspects of the planning. One of these examples is the parallel variant of the insertion algorithm. We introduced this variant in Section 3.3. In the third phase of the approach, k-opt exchanges are used to improve the solution. In k-opt exchanges, we delete k connections in between orders and evaluate all possible other combinations of connections. Russell (1995) is one of the authors who used the three phase local search approach. He uses the seed order generation procedure of Fisher and Jaikumar to select the seed order. In the second phase, Russell uses three ordering rules to select the orders that will be inserted to the route. The three criteria are: earliest time window, farthest distance to the depot, and width of the time window in relation to the distance from the depot. Finally, the solution is improved by deleting and reinserting four order points close to each other. Also Bräysy (2002) developed a three phase local search approach. He creates an initial solution by using one of his two construction heuristics. These construction heuristics are adjustments to the insertion algorithm and the savings algorithm. With help of an ejection chain-based approach an effort is made to reduce the number of routes in the second phase. Also Bräysy uses or-opt exchanges to minimize the total distance.

In literature, we find a lot of different ways to find the best solution for problems with time windows where clustering plays an important role. Some of the algorithms described in Section 3.4.1 can be categorized as metaheuristics, for example the extensions to the algorithm of Fisher and Jaikumar. Other examples are metaheuristics that focus more on the relationship between routes and nodes. In those heuristics, the seed orders are selected according to the differences in time windows or so that they are geographically as dispersed as possible with regard to a previously chosen seed order with the largest number of orders between them (Liu & Shen, 1999). Feo and Resende (1995) had a similar way of thinking. They developed the greedy randomized adaptive search procedure (GRASP). This procedure selects its seed order based on their geographically most dispersed seed orders or the most time constrained seed orders. Another class of heuristics which has a high incidence in literature are the heuristics based on tabu-search. Tabu-search works with an initial solution, which is improved by looking at the neighborhood solutions. A solution is only accepted if it is the best of all neighborhood solutions that are not evaluated in the previous runs. The number of previous runs that are considered is determined at the beginning of the algorithm. A so called tabu-list, on which the previous tried solutions are remembered, prevents to get stuck in a local minimum. When comparing different metaheuristics, most of the heuristics which use tabu-search score relative good on both solution quality and computational time. Also in tabu-search heuristics, the sequential insertion method has the highest incidence to be used as initial solution.

3.5. Seed selection methods

Although a lot of heuristics use seed orders, seed selection is not a subject which is discussed extensively in literature. Literature in which seed selection is discussed in more detail distinguishes two types: manual seed selection and automatic seed selection. When using manual seed selection, the planner chooses the orders which are used as seed. An advantage of manual seed selection is that the planner has more influence on the outcome of the heuristic, since the heuristic starts at the seed. A disadvantage is that it costs the planner time to select those seeds. When using automatic seed selection, a method is used to choose the orders that are used as seeds. There are different algorithms to select those.

The sequential insertion algorithm uses the most difficult order as seed. Whether an order is difficult is determined by a characteristic of the orders. The order farthest away of the depot is the most common definition of most difficult order, according to literature. However, other selection criteria are possible, for example the order with the smallest service window, the earliest deadline, the earliest latest allowed arrival, or the largest order (Dullaert, Janssens, Sorensen, & Vernimmen, 2002; Joubert & Claasen, 2006; Poot, Kant, & Wagelmans, 2002). The exact results for using these selection criteria are not known. In most cases, only results are given over the whole algorithm. Since the datasets and the execution of the algorithm differ in the cases, it is not possible to make a fair comparison.

The sweep algorithm and the algorithm of Fisher and Jaikumar focus on the geographical location of the order. We introduced both algorithms in Section 3.4. After they clustered orders based on their geographical location, Fisher and Jaikumer (1981) determine the center of each cluster. The location of the center is set as seed. This method gives good results on clustering, but again comparing the results of solely the seed selection step is difficult.

Section 3.4 describes the problem that the cones are oblong when using the sweep algorithm or the algorithm of Fisher and Jaikumer. This behavior can be avoided by using the circle covering method. This method determines for each order the smallest circle with the center at this order, such that the total demand inside the circle is close to the vehicle capacity (Savelsbergh M. W., 1990). The orders are ordered by increasing radius. The order with the smallest radius is selected as seed. An advantage of this method is that the clusters do not depend on the location of the depot. A disadvantage is that the computation time can be become quite large.

Some authors take the result of a traditional heuristic as point of departure. For example, Potvin and Rousseau (1993) use such technique. We already described their approach in Section 3.3.

It is also possible to use a combination of manual and automatic seed selection. The zones need to be determined manually, however after that the seeds are determined automatically. The capacitated clustering problem uses this approach. This problem is briefly introduced in Section 3.4.1.

3.6. CONCLUSION

In this chapter, we answered the question "*What is known in literature about vehicle routing with time windows?*". First, we contrast the algorithms that are currently supported by TRP: the sequential insertion algorithm and the savings algorithm. A strong point of the savings algorithm is that it scores better on visual attractiveness by avoiding trip crossings. This is because the orders are assigned simultaneously to trips, where the orders are assigned sequentially in the sequential insertion algorithm. Minus point of the savings algorithm is that the distance is the only assignment criterion. This can lead to longer waiting times.

We found multiple studies in which is tried to improve the sequential insertion algorithm by making an adjustment to the algorithm. The most researches focus on another assignment method for the orders to the route. For example, Iaonnou et al. (2001) selects the order with the lowest impact with as most important criterion the time windows and inserts this order into a route.

Next to the sequential insertion algorithm and the savings algorithm, a lot of other algorithms that focus on clustering are available in literature. Most of these algorithms are parallel algorithms. We discussed the sweep algorithm, the algorithm of Fisher and Jaikumar, and the algorithm of Bramel and Simchi-Levi. The sweep algorithm works with polar coordinators angles and therefore it is harder to use multiple rings. In the algorithm of Fisher and Jaikumar this is possible, but it is more complicated to consider time windows in this option.

To use time windows in the algorithm of Fisher and Jaikumar, we need a two phase approach. In the first phase, we use the original algorithm. However, since time windows are not considered, some orders area not assigned to a route. In the second phase, we must use another (traditional) heuristic or again the algorithm of Fisher and Jaikumar to plan the orders that are not assigned yet.

Baker & Schaffer (1986) proved that a good initial solution is important. In the literature about local search algorithms, the sequential insertion algorithm and the algorithm of Fisher and Jaikumar are most used to generate an initial solution. Furthermore, we discussed some metaheuristics. The sequential insertion algorithm has a high incidence as algorithm to be used to generate an initial solution.

From the traditional heuristics, the sweep algorithm has the best score. This algorithm works with cones. However, the shape of the scones may cause problems. By using multiple rings this problem can be overcome. Another option is to use zones.

In Section 3.5, we discussed literature about seed selection. It is hard to determine the quality of the seed selection method. There are multiple ways to select automatic seed orders. Changes can be made in the selection criteria of the seed order. Instead of the order farthest away from the depot, also among others the order with the smallest time window may be used. Seed orders can also be chosen based on their geographical location by using a sweep algorithm or predefined clusters.

4. DATA ANALYSIS

In Chapter 2, we described the current situation at Zeeman. In this chapter, we analyze the plans provided by Zeeman to find the causes of the problems mentioned by the planners. In Chapter 5, we use the conclusions of this chapter to find a suitable approach for the planning process that improves the generated plan. In Section 4.1, we describe the problems mentioned by the planners. The data analysis itself consists of three steps. First, we investigate whether improvements are possible by comparing the plan made by TRP with the plan adjusted by the planners of Zeeman (Section 4.2). Section 4.3 discusses the effect of the improvement steps. Next, we investigate the cause of the problem (Section 4.4). We analyze all cities which are visited by multiple trucks and divide the problem causes into multiple categories. The chapter is finishes with a conclusion in Section 4.5.

To investigate why orders are not clustered in the current plan made by TRP, Zeeman provided three cases for analysis. For the data analysis, two plans per case are used. The first plan is the plan generated by TRP. The second plan is the same plan, but with the manual adjustments by the planners of Zeeman included in order to better accommodate their preferences.

4.1. PROBLEMS

Chapter 1 addresses two main differences in the optimization criteria between Zeeman's planners and TRP. In Section 4.1.1 and 4.1.2, we explore these problems in more detail.

4.1.1. CITIES

The first finding of the planners of Zeeman is that multiple trucks supply stores in the same city while it should be possible to deliver those stores within the same city with fewer trucks. According to the planners, it costs extra time to supply stores in the city center. This is something TRP does not consider. TRP can only make the difference between roads the built-up area and outside the built-up area.

Figure 17 and Figure 18 show an example of the Dutch city Breda. Figure 17 shows the route as TRP generated it. Two trucks visit Breda. One truck enters the city center on Tuesday during day time and one truck enters the city center on Wednesday during day time. Figure 18 shows the situation after the planners of Zeeman manually adjusted the route. In the solution of the planners of Zeeman, one truck enters the city center of Breda during daytime on Wednesday and delivers the orders of both TRP's trucks destined for the city center. Of course, this change has impact on the rest of the planning. After all, the stores outside Breda, which were originally assigned to the same truck, need to be planned in an additional truck. However, according to the planners of Zeeman this leads to a cheaper overall solution. We investigate this claim in Section 4.2.

4.1.2. CAPACITY

The problem in Section 4.1.1 is on a city level. The planners of Zeeman also claim that on region level improvements in clustering are possible. In some cases, a truck drives to a region far away from the depot. One of the orders is then delivered during the ride to this region or back from this region to the depot. According to the planners, this is inefficient use of the truck's capacity. Driving to a region is the most expensive part of a trip. If the planners have to choose between delivering an order on the way to the region and sent an additional truck to the region to deliver the remaining orders, or deliver all orders in a region by one truck and sent an additional truck to a region that is closer to the depot, they will choose for the latter. We clarify this by an example in Figure 19 and Figure 20. These figures show the region west of Paris in France. The red line in Figure 19 shows a trip which first delivers an order in Belgium, than drives almost 400 kilometers, and subsequently delivers two orders in France. Figure 20 shows that the planners of Zeeman choose for a different solution. The order in Belgium is delivered in the trip depict with the blue line, while the two orders in France are delivered in the trip depicted with the yellow line. This difference is apparent in the two graphics, with the Zeeman planning visually appearing more attractive.

FIGURE 17 - BREDA IN TRP SOLUTION

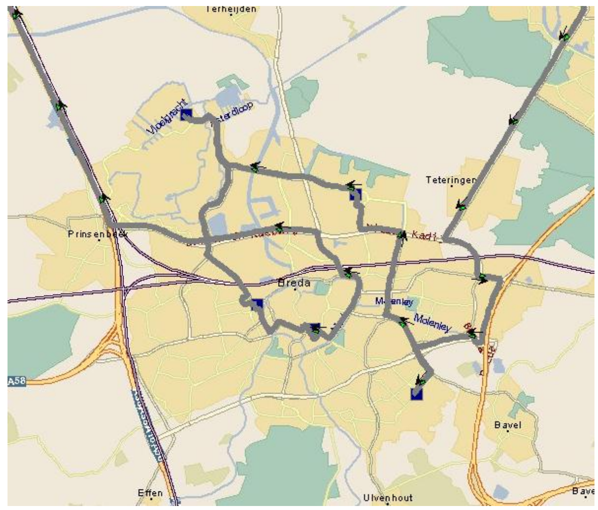


FIGURE 18 – BREDA IN ZEEMAN SOLUTION

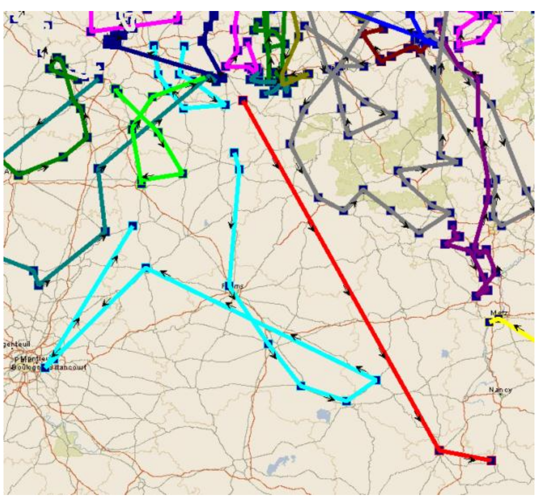


FIGURE 19 - EAST OF PARIS IN TRP SOLUTION

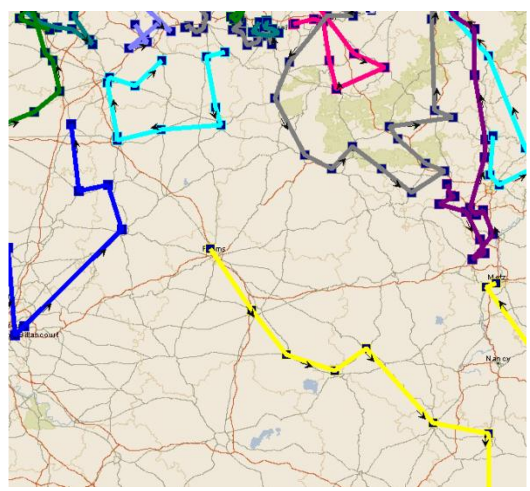


FIGURE 20 - EAST OF PARIS IN ZEEMAN SOLUTION

Related to the problem that driving distance between two deliveries is large is the observation of the planners that the capacity of the truck is not fully utilized. Especially when the orders that are delivered by this truck are further away from the depot, the planners believe that it is a waste to not utilize the capacity of the truck. They also believe that by combining trips, a cheaper solution is possible. In the example of Figure 19, we see that the trip with the red line transfers 10 trolleys while truck has a capacity of 46 trolleys. The trip depicted with the blue line delivers 40 trolleys. In the solution of Zeeman (Figure 20) the trip depicted with the blue line delivers 46 trolleys and the trip depicted with the yellow line 43. The utilization of the capacity of the trucks seems better for the solution of Zeeman. However, we have to further investigate this to find out what the effect is for the whole planning. We perform this analysis in Section 4.2.

4.2. Comparing the plans

The first step in the data analysis is to confirm that improvements in the planning are possible. We compare a plan that is manually adjusted by the planners of Zeeman with the planning TRP generates with the same input, so before the manual adjustments are done.

4.2.1. Performance indicators

To compare the plan made by TRP with the plan made by Zeeman, we define performance indicators. Although most authors use performance indicators to evaluate their work, there is no standard set of performance indicators available. Consequently, we are left with the need to define our own.

In this research, we distinguish two kinds of indicators. The first set of indicators is a collection of the most used indicators in literature that are relevant for this research. These indicators say something about the quality of the solution in general. Some examples of indicators are the number of orders planned, the cost of the planning, and the total driving distance. The second set of indicators says something about the quality of the solution in relation to the problems defined in Section 4.1. We first determine which cities are visited by more trucks than required. For this purpose, we consider the total number of trolleys that need to be delivered in a city, the time windows of the stores in that city, and the vehicle preferences for an order. With this information we, we solve a set covering problem. This results in a lower bound for the number of trucks that need to visit a city. It is a lower bound, since not all factors are considered, such as limitations on driving durations. We say that a city is visited more often than required if the number of trucks that visits the city in the plan is higher than the lower bound for that city. Furthermore, we calculate the radius of the trip. The radius is the distance in a straight line between the two orders in the cluster that are farthest away from each other. This indicator gives us some insight in the density of the clusters. To get some insight in the capacity utilization of the trucks, we calculate the average, maximum, and minimum utilization for each plan. The capacity utilization is calculated by dividing the number of trolleys planned in a vehicle by the capacity of that vehicle. Furthermore, we calculate the total number of kilometers driven between the first delivery and the last delivery in a trip and the number of kilometers driven between two orders.

Zeeman provided three different plans to analyze in this research. The difference between those three plans is the total number of trolleys that are transported, respectively roughly 3800, 4200, and 4600 trolleys. After a first brief analysis of all three plans named in the paragraph above, we conclude that the proportion between the performance indicators of the different plans is equal. Therefore, the conclusions we draw from this analysis are equal for each plan. We only discuss one case in this chapter. We choose to show the plan of 4200 trolleys, since in that case the difference between the plans of Zeeman and TRP are most clear. We summarize the most interesting performance indicators in Table 2 and Table 3. The full tables are in Appendix B and Appendix C.

Table 2 - General performance indicators, Zeeman vs TRP

Indicator	Zeeman	TRP	Difference	%-difference
Number of orders	970	970	0	0%
Number of transported trolleys	37,115	37,115	0	0%
Number of trucks used	85	96	-11	-11.5%
Number of kilometers driven	41,605	50,549	-8,994	-17.7%
Number of driving hours	687	827	-140	-16.9%
Costs	284,496	327,657	-43,161	-13.2%

Table 3 - Specific performance indicators for the automatically planned vehicles, Zeeman vs TRP

Indicator	Zeeman	TRP	Difference	%-difference
Cities where more trucks visit than required	15	62	-47	-75.8%
Average number of kilometers driven between first and last order	285	337	-52	-15.2%
Average distance to next order	24.6	37.0	-12.4	-33.5%
Average radius of clusters	0.76	1.01	-0.25	-24.6%
Average capacity utilization	93.0%	84.0%	9.0%	10.7%
Maximum capacity utilization	102.2% [1]	100.0%	2.2%	2.2%
Minimum capacity utilization	67.4%	6.5%	60.9%	933.3%

From Table 2, we conclude that on all points the planning adjusted by the planner scores at least as good as or better than the plan made by TRP. An important difference is in the number of kilometers driven and the number of driving hours. These are both much lower in the plan made by Zeeman. The results in Table 2 show that improvements are possible.

The planners of Zeeman suspect that the difference in quality between the plans is caused by the number of trucks that visit a city. Table 3 shows the specific performance indicators. Indeed, there is a large difference between the number of trucks that visit a city in the plan made by Zeeman and in the plan made by TRP; respectively 15 and 62 cities. In Section 4.3, we further investigate this finding. Also the radius of the cluster is almost 25% reduced by the planners.

The average number of kilometers driven between the first and the last in a trip order differs 51 kilometers per trip between the solution made by TRP and the solution made by Zeeman. The kilometers driven from or to the depot are lower in the plan made by TRP. However, the total number of kilometers driven is higher in TRP and more vehicles are used. Even though, TRP uses more vehicles, the average number of kilometers driven per shift is higher. The average distance to the next order in a trip is also higher in the TRP solution. This can either mean that the route of the vehicle is not optimal or the orders are not optimally assigned to the vehicles. We random picked

[1] The capacity of the truck is an estimation of the number of trolleys that fit into a vehicle. The planners know from experience, that it is in some cases possible to transport more than 46 trolleys with a vehicle. In the planning adjusted by the planners, there are six vehicles that transport more than 46 trolleys. Therefore, the maximum capacity is higher than 100%.

some trips and tried to optimize the sequence in which the orders are planned. In all selected trips, this was not possible. With this finding and the findings earlier in this thesis, we conclude that improvement in the average number of kilometers driven between the first and the last order in a trip can be made by assigning orders in another composition to the vehicles.

Over the whole plan, the planners of Zeeman do a better job with respect to the capacity utilization. When we compare the separate utilization for the shift work vehicles and the short multiple day vehicles, the differences appear (Table 4). We find that in particular the average utilization of the shift work vehicles is much lower in the plan made by TRP.

TABLE 4 - CAPACITY UTILIZATION FOR ZEEMAN

Indicator	Zeeman - Shift work	TRP - Shift work	Zeeman - Short multiple day	TRP - Short multiple day
Average capacity utilization	92.0%	76.2.0%	93.7%	89.7%
Maximum capacity utilization	101.7% [1]	100.0%	102.2% [1]	100.0%
Minimum capacity utilization	75.0%	6.5%	67.4%	16.5%

Figure 21 shows the distribution of the capacity utilization for all shift work vehicles and short multiple day vehicles. We find that in the plan made by TRP there is high peak round 100% capacity utilization. Furthermore, we find that in the plan made by TRP there are more vehicles with a very low capacity than in the plan of Zeeman. On the other hand, the planners of Zeeman planned more trolleys than the capacity is available on six trucks. This is possible due to the intimate knowledge of the vehicles by the planners, because they know that in some cases one or two extra trolleys will fit into the vehicle. This information is unknown to TRP and therefore TRP cannot use this information in optimizing the planning. Based on the distribution, we conclude that TRP tries to plan the vehicles as full as possible. This results in a high number of vehicles with an utilization close to 100%, but also a higher number of vehicles with a worse utilization. Zeeman balances the work load more over the vehicles and therefore scores higher on the average utilization.

4.2.2. COMPARISON OF THE INDIVIDUAL ROUTES

From Section 4.2.1, we conclude that improvements in TRP are possible. From the specific performance indicators, we conclude that a reduction of the number of trucks that visit a city and a minimization of the driving distance between orders in a trip leads to a more clustered solution. The capacity utilization seems less promising as method to improve the planning. The question remains why the planners of Zeeman are able to improve the planning and TRP cannot. Having compared the full plans using performance indicators in Section 4.2.1, we now turn to the comparison the individual routes.

The first thing that we notice when comparing the routes of the planning is that most routes are changed by the planners of Zeeman. Only 21 vehicles transport the exactly same set of orders in the plan of TRP and the plan of Zeeman. It may be that a single order is added to or removed from the vehicle, but also the whole composition of orders may be changed. This large amount of changes

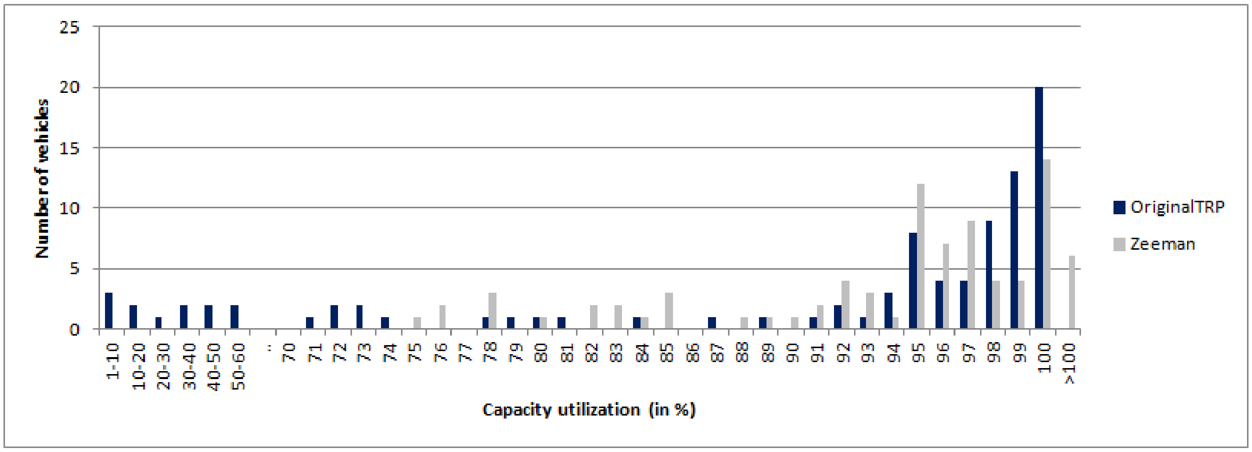


FIGURE 21 - DISTRIBUTION OF THE CAPACITY UTILIZATION

and the size of the plan make it difficult to compare the individual routes in both plans. Therefore, we do not compare the routes in the different plans. In Section 4.3, we exam the individual routes of the plan of TRP.

In Section 4.1, we have listed two problems the planners experience when evaluating the planning. When we compare the plan made by Zeeman with the plan made by TRP it is evident that the planners have adjusted their planning accordingly. From the performance indicators, we conclude that the number of trucks that visit a city is reduced. In addition, in graphical depictions of the planning, we find that most routes are more compact; in the plan made by Zeeman the routes are centered around cities, whereas in the plan made by TRP contains repeated visits to the same city by multiple vehicles.

4.3. THE EFFECT OF THE IMPROVEMENT STEPS

Although, the solution after the improvement steps solution is the solution which is examined, we do not need to underestimate the importance of the initial solution. In Chapter 3 we explained that problems with the best initial solution give the best overall result. We generated a plan without running the improvement steps. This is the initial solution. We use the term final solution for the solution after the improvement steps the final solution. In this section, we contrast the initial solution and the final solution.

The performance indicators of initial solution are in Table 5 and Table 6. The costs are reduced by 2.2% from the initial to the final solution. The number of kilometers driven is reduced by more than 5%. Again, we focus on the number of cities that are visited by more trucks than necessary. We find that in the initial solution the number of cities visited by more trucks than required is reduced by seven trucks. In 51 of the total 69 cities the situation does not change. The orders in those cities are still delivered by the same vehicles. There are 22 cities that are visited less times after the improvement steps. However, there are also eleven cities in which the situation got worse, while the overall planning got better. When we investigate this last group of cities, we find that in most cases these trips cover a large distance. In other words, the orders are not clustered all around a city or region. Furthermore, in most of the cities, the orders are scattered over the city and not clustered in the center. The improvement steps have no specific defined incentive to cluster orders in one city, but they do want to reduce the costs, and thereby the number of kilometers driven and driving time, as much as possible. To verify the behavior of TRP, the manually adjusted plan of Zeeman is taken as initial solution and we run the improvement steps. The improvement steps do not find any improvement. In conclusion: we find that the initial solution is the most fruitful area of improvement, which coincides with the focus of this research.

TABLE 5 - GENERAL PERFORMANCE INDICATORS, INITIAL VS FINAL

Indicator	Initial	Final	Difference	%-difference
Number of orders	968	970	2	0.2%
Number of transported trolleys	37,058	37,115	57	0.2%
Number of trucks used	97	96	-1	-1.0%
Number of kilometers driven	53,214	50,549	-2,665	-5.3%
Number of driving hours	863	827	-36	-4.4%
Costs	334,941	327,657	-7,284	-2.2%

TABLE 6 - SPECIFIC PERFORMANCE INDICATORS FOR THE AUTOMATICALLY PLANNED VEHICLES, INITIAL VS FINAL

Indicator	Initial	Final	Difference	%-difference
Cities where more trucks visit than required	69	62	-7	-13.7%
Average number of kilometers driven between first and last order	365	336	-28	-8.5%
Average distance to next order	38.1	37.0	-1.1	-2.9%
Average radius of clusters	1.16	1.01	-0.15	-14.5%
Average capacity utilization	84.8%	84.0%	0.8%	0.9%
Maximum capacity utilization	100.0%	100.0%	0%	0%
Minimum capacity utilization	9.3%	6.5%	2.8%	43.3%

To answer the question why the planners of Zeeman are able to improve the plan made by TRP, we look at the algorithm TRP uses. This algorithm has been introduced in Chapter 2. In TRP, the improvement steps are meant to make a good solution, the initial solution is to generate a feasible plan as point of departure. A disadvantage of the improvement steps is that for each step a solution is only accepted if it yields a direct improvement for the whole planning. When we compare the plan of Zeeman and TRP, we still find some cases where multiple steps are needed to improve the solution. There are three main reasons why the improvement steps cannot find this solution. The first reason is that more iterative steps are needed to find this solution. Second, it may be that multiple moves or exchanges are necessary to retrieve a better solution. However, when the first move that is needed is not profitable for the whole planning, this solution will no longer be considered. A planner sees that making that first non-profitable step will have benefits later in the planning, perhaps because he can judge the planning visually. Third, the batch of orders that may need to be moved to another vehicle is too large. Moving one of the orders to another vehicle will not be profitable, however moving half of the orders to another truck will. This is again a solution

TRP will not consider. Since orders are moved or exchanged one by one, too many steps need to be made to let TRP find the solution the planners of Zeeman find.

ORTEC wants to investigate whether an improved initial solution leads to a better final solution. The improvements steps will still play an important role. However, the focus will be on clustering in the initial solution.

4.4. Analysis of the cities

We perform data analysis to find the reason why TRP does not cluster the orders. TRP plans deliveries to 970 stores of Zeeman in 755 cities. There are 195 cities with two or more Zeeman stores. There are nine cities which are visited by more trucks than required in both the plans of Zeeman and TRP. The number of trucks that visit the city is equal in both plans. For these cities, we conclude that the lower bound is not realistic.

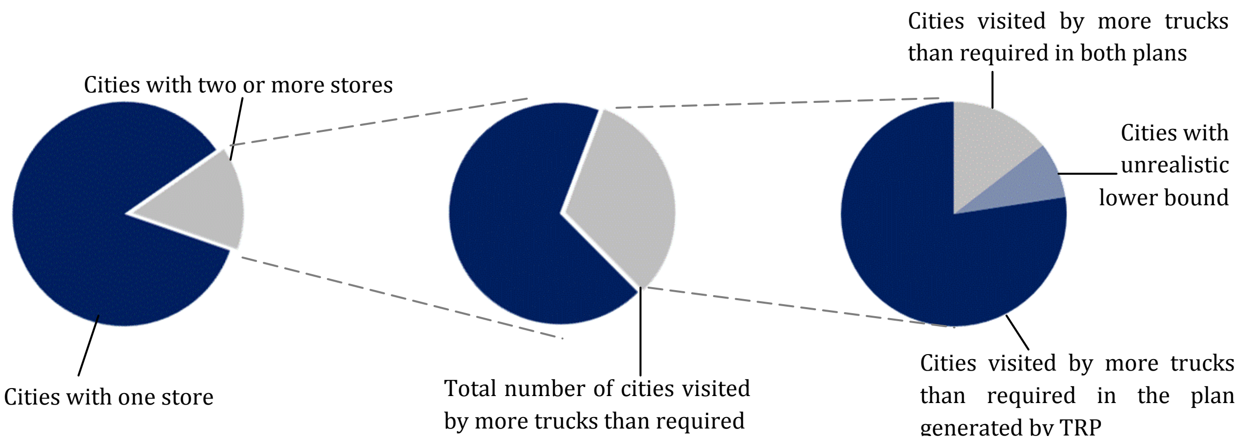


Figure 22 - Pie charts of distribution cities visited by more trucks than required

There are also five cities which are visited more times than the lower bound in the plan of Zeeman. However, in TRP the number of trucks that visit that city is higher than the number of trucks that visit that city in the plan of Zeeman. These five cities are larger cities which need to be supplied with two trucks or more. Due to the multiple trips through this city, it is complex to retrieve the cause of the multiple visits. Therefore, we do not consider these cities in our analysis.

In Gent, one additional truck visits the city in the plan made by Zeeman in comparison to the plan made by TRP. Figure 23 and Figure 24 show the different routes for Gent. The planners deliberately change the truck and therefore choose to not merge all orders in that city to one truck. Since it is feasible to plan all orders in one truck, we cannot change this behavior in TRP, unless we program the way of thinking of the planner. Since this is exception, we do not think this is useful and necessary.

There are 48 cities in the plan of TRP where the number of trucks that visit the city is higher than the lower bound and higher than the number of visits in the plan of Zeeman. Since the planners of Zeeman succeed in visiting those cities less times, we conclude that another feasible solution is possible. For these 48 cities, we try to find out why multiple trucks visit these cities while this is not necessary.

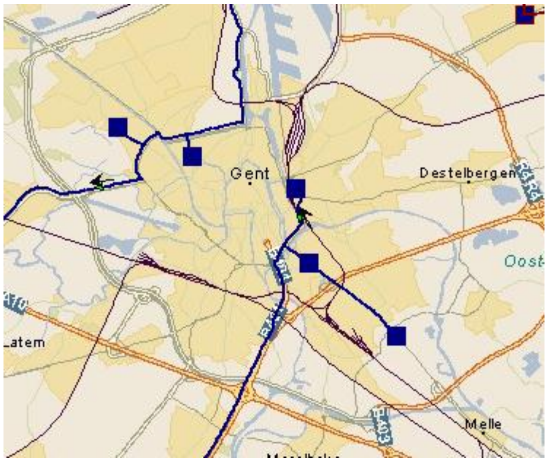


FIGURE 23 - GENT IN TRP

FIGURE 24 - GENT IN ZEEMAN PLANNING

4.4.1. IDENTIFYING CAUSES

We find out why trucks unnecessarily visit the city more times than required by analyzing the trips related to the cities identified in the previous section. We prefer to evaluate the final solution. However, due to the changes by the improvement steps, it is difficult to retrieve the cause. From Section 4.3, we know that 51 cities that are visited by more trucks than required in the final solution are already visited by too many trucks in the initial solution. We only consider the cities that are visited by more trucks than required in both the initial and the final solution. This are 40 cities. The initial solution uses the sequential insertion algorithm, which we can reproduce for a trip. This comparison is not totally fair, since we do not consider the sequence in which the trips were generated; it could be that an order was already planned in a trip and therefore could not be assigned to the preferred trip. However, this analysis still gives us an idea about the cause of the problem.

We define four categories of what could be the cause of the problem. These categories are based on the restrictions of trip, as currently defined in TRP.

CAPACITY OF THE TRUCK REACHED

When using the sequential insertion algorithm, the route is for a large part determined by the seed. For example, a route can completely change even though the seed order changes to a neighboring order. It could be the case that a truck starts in City A and Route I emerges. At a certain point there is enough capacity left in the truck for exactly one order. The nearest insertion is an order in City B. However, in City B there are multiple orders. The sequential insertion algorithm inserts the first order in that city in Route I. The next order City B will probably be the first order in a new route. When an order in City B was taken as seed, the orders would probably be planned in the same truck. The selection of the seed can be seen as main cause for this category.

In TRP, we perform some test. We random select some orders as seed order and let the sequential insertion algorithm assign orders to the trip. We conclude that it depends on the chosen order as seed order whether the orders are planned in one trip. If the seed is in or close to a city with multiple orders, we find that in almost all cases the orders in the city are planned in one trip. In Chapter 5, we investigate how we should define a good seed order.

TIME WINDOWS

The time windows could be a reason why two orders in the same city are not planned in one truck. If there are two orders in a city of which one can only be delivered during day time and one only during night time two vehicles should enter the city. The time windows of the orders are considered when we calculated the lower bound for the number of vehicles and therefore not counted in the performance indicator. However, it can also be the case that there are two orders in a city of which one order can only be delivered during night time, while the other order has no time window. If the current truck is a truck with a shift during day time the order without a time window can be added to that vehicle. The other order is not feasible in this shift and will therefore be inserted into another vehicle. When the shift is a night time shift there would be no issue and both orders will be planned in the same trip. The vehicle selection influences the seed choice. An unfortunate vehicle choice may lead to multiple visits of this city.

In TRP, we perform some tests. Indeed we find that when we choose another vehicle, the orders mentioned in the previous paragraph are planned in one trip. We can conclude that we cannot blame the assignment method, but should consider the relation between the vehicle choice and the seed choice.

ORDER IN NEIGHBORING CITY CLOSER

The sequential insertion algorithm assigns the order closest to the seed to a vehicle. This is not necessarily an order in the same city. For example, we have a seed order that located in the North side of a city. The algorithm searches for the next order to insert and finds two possible (and feasible) candidates. The first candidate is an order in the same city, but in the South. The second candidate is an order in a neighboring city. However, this neighboring city is located north of the initial city. The second candidate order is closer to the seed order. Therefore, that order is inserted into the route. The assignment method used in the sequential insertion algorithm is the cause of this category. If an order is a neighboring city is closer, it is questionable if it is desired to force the algorithm to plan the order in the same city in the trip instead of an order in a neighboring city. Anyway, a different seed choice would lead to other assignment considerations and therefore plans the orders in the same city.

In TRP, we perform some tests with orders that are in the same city, but where there is also an order in a neighboring city. We conclude that it depends on the seed choice, whether the orders in the city are delivered by one or two trucks. Therefore, we cannot solely blame the assignment method for this cause, but also should consider the seed selection.

REQUIRED VEHICLE TYPE

Each order has one or two vehicle types with which the order should be delivered. In case of Zeeman, the required vehicle type is mainly based on the geographical location of the order. Therefore, it is not very likely that two orders in one city have different vehicle requirements. If two orders in a city do have different vehicle type requirements, we considered this while calculating the lower bound. In an exceptional case, it is possible that one order in a city can be delivered by two vehicle types, where the other order only can be delivered by one of those two vehicle types. In that case, we create a similar situation as with the time windows: it is possible that due to an unfortunate vehicle choice two orders in one city are not planned in one trip. Although, we expected a low occurrence of this cause, we do consider this as option.

4.4.2. Assigning cause

We evaluate the 40 cities that have a higher number of visits than required, but have the required number of visits in the plan made Zeeman. We analyze the problem per city. Table 7 gives the occurrences of the categories.

Table 7 - Occurrences of categories

Category	Occurrence
Capacity of the truck reached	10
Time windows	11
Order in neighboring city closer	9
Required vehicle type	0
Unclear cause	10

From Table 7, we conclude that four categories have a high occurrence. Only the required vehicle type is not found as cause of the multiple visits of the cities. In Section 4.4.1, we already explained that we expected this result. Furthermore, we found eight cities in the category 'unclear cause'. These are cities of which we could not identify the cause of the multiple visits. Some of those cities, we could assign to two categories. For example, the capacity of the truck was reached, but the time window of the order would also not allow adding that order to the current trip. For other cities, we were not able to identify a cause.

The frequency of occurrence for the other three categories does not differ much. This means, we cannot assign just one cause, but should look at a broader picture. All of these categories can be improved by choosing a seed in a more considered way. For all categories, an incentive to keep orders in the same city in one truck is missing. Especially in the case of the neighboring order, it can be questioned if it is a desired situation to have this incentive. If splitting the orders in two trucks leads to a cheaper planning, this may be a better solution. However, with the planning in Zeeman is proven that this is not necessarily the case even ignoring higher costs of drive to an order slightly further away.

4.5. Conclusion

In this chapter we answered the question "*Why is the visually attractiveness lower in the current planning of TRP?"*. We described two points the planners are discontent with in the current planning of TRP. The first problem is that some cities are visited by more trucks than required. The second problem is that the capacity of the truck is not used optimally to the way of thinking of the planners. We showed that the planners of Zeeman indeed make improvements to the plan made by TRP. We defined a number of performance indicators to find out whether these problems are the cause of the improvements made by the planners. Indeed, the number of cities which is visited by more trucks than required is lower in the plan made by Zeeman. The first order is delivered earlier in the route, sometimes on the way to a specific region. According to the planners, it is more efficient to drive with a full truck to that region. The capacity utilization does not differ much between the plans. Furthermore, we compared the routes of the two plans. We found that most of the routes of TRP are changed by the planners.

Subsequently, we compared the initial solution with the final solution. We found that the initial solution is improved. However, no large improvements on clustering are obtained. The remaining question is why the planners can improve the solution, but TRP not. The answer lies in the way the improvement steps work. Since they operate step by step, they do not consider the consequences over multiple steps. This initiation is a quality human beings have over deterministic algorithms.

We searched for the reason some cities are visited by multiple trucks. We defined four categories with possible causes. There was no univocal cause found. The categories ‘capacity of the truck reached’, ‘time windows’, and ‘neighboring city closer’ occurred almost with the same frequency. Underlying main cause for all categories is the selection of the seed. We conclude that the way the method assigns the orders to the trips depends strongly on the seed choice. However, with the right seed choice the assignment method works as expected.

Finally, we found that the most important change the planners of Zeeman brought is more clustering of the orders. This means that the delivery of orders is more focused on one region and let as few trucks as possible visit a city.

5. Approach

This chapter assembles the parts contributed by the previous chapters and presents a solution approach. In Chapter 2, we described the way TRP currently works and the characteristics of the planning in detail. In Chapter 3, we explored literature to find possible directions for the solution of our problem. In Chapter 4, we searched for the cause of our problem. In this chapter, we answer the question "*How can the algorithms found in literature be adapted such that they become applicable to improve the planning of clients of ORTEC, such as the planning of Zeeman?*".

Before answering this question, it is instructive to go back to the goal of the algorithms found in literature. In essence, an initial solution constructs for each vehicle a set of orders to be delivered by that vehicle from the available individual orders and determines the sequence in which these orders are delivered. Consequently, it is of interest which individual orders are chosen to be combined (tying in with incentives and feasibility) and how this combination is made within the larger context of the algorithm (relating to the different approach found in literature). These two parts shape the flow of this chapter, which we further detail below.

We briefly repeat the goals we try to achieve with the new approach. The main goal of this research is to make a better planning in TRP, such that the planners need to make less manual adjustments. The planners have two additional whishes they prefer to see in the planning. First, each city should be visited with as few trucks as possible. Furthermore, the planners prefer that all orders in a vehicle are relatively close to each other (Chapter 4). ORTEC also has some wishes for the solution. They prefer to keep the changes to the current solution as small as possible. In that way, it is easier to integrate the algorithm in the current software of ORTEC and it is easier from a support point of view. Furthermore, they like to see the adjustments in the algorithm of the initial solution and not in the improvement steps.

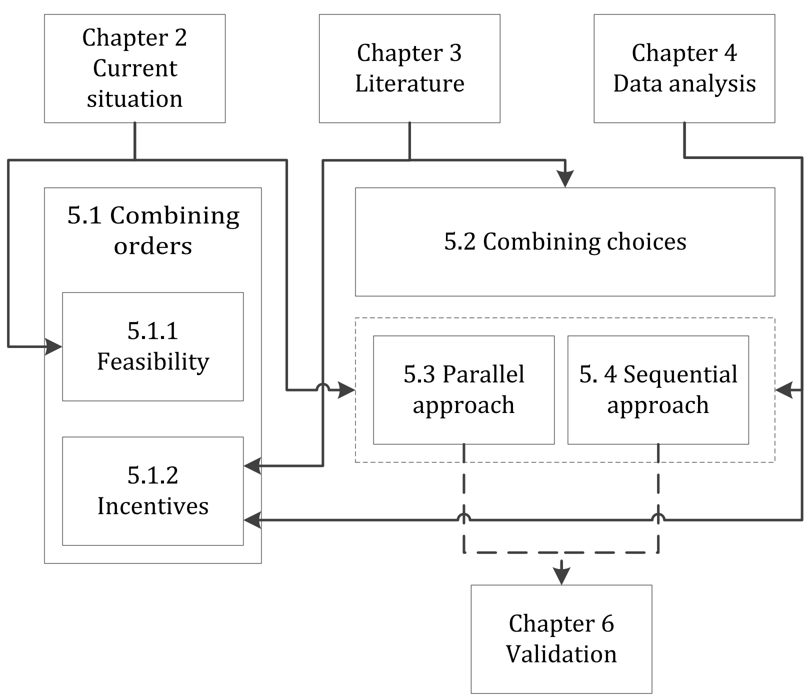


FIGURE 25 – CORRELATION BETWEEN THE DIFFERENT PARTS OF THIS CHAPTER

This chapter is structured as depicted in Figure 25. Section 5.1 examines the choice of orders to be combined, looking consecutively at feasibility and incentives for such a choice. To define the feasibility, we use the characteristics of the planning defined in Chapter 2, while Chapter 3 and 4 are used to shape the incentive part of the choice. Subsequently, Section 5.2 deals with the combination of these choices, by dividing the algorithms found in literature (Chapter 3) in three groups, two of which are covered in this research. For these two groups, we provide an overview of their approach and discuss differences. Furthermore, we briefly discuss the dismissed approaches in this section. Having covered the common ground, we describe the two approaches, parallel and sequential, in detail in Section 5.3 and 5.4. We end the chapter with conclusions in Section 5.5.

5.1. COMBINING ORDERS

Generating an initial solution focuses primarily on creating a feasible plan. In Chapter 2, we described the characteristics of the planning of Zeeman. We use these characteristics to define the meaning of feasibility in the context of this research. This research focusses on improving the initial solution of TRP's planning algorithm with the purpose of increasing clustering. So in addition to requiring a feasible solution, we need to define an incentive that focuses on clustering. Both, feasibility and incentives influence the combination of orders; feasibility restricts the choice to those orders resulting in a feasible combination, while incentive promotes the choice most appropriate for a clustered solution. Both are covered in a separate section. In Section 5.3 and 5.4, the concepts of Section 5.1 are adapted and specified where needed.

5.1.1. FEASIBILITY

In the context of TRP's planning algorithm, an initial solution consists of a number of clusters, where the delivery of all order in the clusters is feasible. We define a cluster as a set of orders that is delivered in one trip, but of which the position of the orders in the trip is not known yet[2]. A trip that has to be executed in real life has a lot of restrictions. The equipment, such as the truck and the trailer, has limitations, the driver is not always available, there are wishes or requirements for the delivery of the orders, and government regulations further restrict delivery by trucks. The number of restrictions to be considered can easily explode. The more requirements a model takes into account while generating the cluster, the more reliable the result. However, when we take all these requirements into account the model becomes very complex and the calculation time becomes large. Therefore, we only consider the most important options in this section. We successively discuss the capacity of the truck, the time windows of the orders, and the duration of the shift as requirements involved in our model.

THE CAPACITY OF THE TRUCK

The total load of all orders in a cluster should be smaller than or equal to the capacity of the truck.

TIME WINDOWS OF THE ORDERS

The time window of the order is a subject that has returned several times in this research. The challenge is to match the time window of the order with an available shift. For each order and each shift, we calculate whether there is an overlap between the time window of the order and the shift.

[2] The reason why the position of the order in the trip is initially unknown will become clear in Section 5.2, when we compare and contrast the parallel and sequential approach.

If that is the case, we say that is possible to deliver the order in that shift. This is not fully consistent with reality, since there are more aspects that influence that decision. The most important one is the driving time to the location of the order. The location of an order can be a couple hours away from the depot. The truck needs to deliver the order before the end of the time window of the order and needs to be back at the depot before the end of the time window of the shift. To avoid this situation, we add the driving time from the depot to the order to the start time of the shift, and subtract it from the end time of the shift. This is defined in Equation 2 and Equation 3.

$$\text{TWstart}_{s,o} = \text{starting time of shift } s + \text{ driving time from the depot to order } o \qquad \textbf{Equation 2}$$

$$\text{TWend}_{s,o} = \text{end time of shift } s - \text{driving time from order } o \text{ to the depot} \qquad \textbf{Equation 3}$$

For the same reasons mentioned for the driving time, we also need to consider the service time. The service time is the time needed to unload the order from the truck. The unloading must be finished before the end of the time window. We update the end of the time window as in Equation 4.

$$\text{TWend}_{s,o} = \text{end time of shift } s - \text{driving time from the order } o \text{ to the depot} - \text{service time of order } o \qquad \textbf{Equation 4}$$

Now, we calculate for each order whether it is possible to deliver it with this shift. That is the case if the answer to one of the equations 5 through 8 is true. Shift denotes start or the end of the time window of shift *s*. Note that it is only possible to check the feasibility of a cluster with respect to the time window if we know for which shift we need to check this.

$$\text{TWstart}_{s,o} < Shift_{\text{start,s}} \text{ AND } \text{TWend}_{s,o} > Shift_{\text{start,s}} \qquad \textbf{Equation 5}$$

$$\text{TWstart}_{s,o} > Shift_{\text{start,s}} \text{ AND } \text{TWend}_{s,o} < Shift_{\text{start,s}} \qquad \textbf{Equation 6}$$

$$\text{TWstart}_{s,o} < Shift_{\text{start,s}} \text{ AND } \text{TWend}_{s,o} > Shift_{\text{end.s}} \qquad \textbf{Equation 7}$$

$$\text{TWstart}_{s,o} < Shift_{\text{start,s}} \text{ AND } \text{TWend}_{s,o} > Shift_{\text{end.s}} \qquad \textbf{Equation 8}$$

A final type of constraint introduced by the time window concept is that of overlapping time windows among orders. Recall that our clusters do not contain any information on the position of the orders within a trip. It is therefore unknown what route is traversed by the vehicle, which in turn defines the actual delivery within the time window for each order. In some cases this position may be important. We explain this with an example. We have two orders. Both orders can only be delivered on Mondays, but have no time restrictions. Both orders are about one and a half hour away from the depot. We calculate whether it is possible to deliver the orders with a shift starting on Monday 22:00. The answer for both orders is that it is possible. However, when we deliver the first order with the shift, there is barely time left to deliver the other order in the same shift. This becomes a problem when the driving time between the two orders is greater or equal than half an hour. This overlap gets more and more difficult to detect when more orders and/or more complex time windows are involved. We have been unable to find solutions that address this constraint without an overly large computational burden. As a consequence, the feasibility check is not complete with respect to overlapping time windows among orders. Due to the fact that Zeeman's planning rarely features small time windows and with the knowledge that a solution would take an inordinate amount of time, we choose to accept this shortcoming of our feasibility check.[3] In Chapter 6, we validate this choice and later list improvement of this check as part of future research.

In Chapter 4, we concluded that the time windows of the orders play an important role in the planning process. Later in this chapter, it becomes clear that the time windows limit the number of possible clusters. A lot of orders cannot be delivered the same shift and therefore, they cannot be in one cluster.

Duration of the shift

The duration of a shift is the sum of the driving time, waiting time, service time, and breaks in that shift. There is a limit on the duration of a shift. This maximum duration depends on the type of shift.

The European Union set legislations on the driving hours (European Parliament, 2006). These legislations ensure that the drivers take enough breaks during their shift. The exact legislations are too complex to discuss here and additionally not in scope of this research. However, they have influence on the duration of the shift, and therefore on the planning. In our case, most of the shifts have a fixed amount of breaks a driver needs to take. This may be a short rest break or a longer night break. We discussed the breaks in Chapter 2. We subtract the time of the breaks from the shift duration.

Both the driving time and the waiting time strongly depend on the route of the trip. Due to the unknown position of the orders in the trip, it is not possible to determine the driving time and waiting time of a trip. However, even without knowing the sequence, we can still estimate the driving time. In other products of ORTEC, the driving time is estimated by calculating the driving time from one order in the middle of the cluster to a number of other orders in the cluster. We call the order in the middle of the cluster the seed order. Figure 26 shows an example of this approach. All other orders within the circle are in the cluster. Order A is the seed order. In this example, we use five orders to estimate the driving time of the cluster. The five orders closest to Order A are Order B, C, E, F, and H. We calculate the driving time from order A to order B and back, from A to C and back, and so on. We use the sum of the driving time of the five trips as estimation for the total driving time in the cluster.

This method has its limitations. For example, the estimation of the driving time is less accurate if the chosen number of orders is relatively close to the location of the seed order, while the other orders in the clusters are relatively far away. Therefore, we also test some alternatives where the following input is used as estimation for the driving times:

- The driving time to a number of orders (say n) farthest away from the seed in the cluster. n is equal for each cluster.

[3] Finally, as we will see later, the effect on the parallel approach is reduced by the improvement steps, while the sequential approach is not affected at all, because the feasibility check is merely used to determine the seed order.

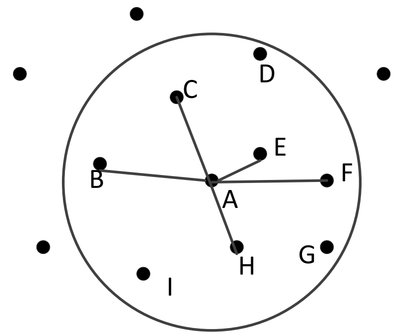


FIGURE 26 - CALCULATE DRIVING TIME

- The driving time to a number of orders (say k) in the cluster. The orders are randomly chosen from all orders in the cluster. k is equal for each cluster.
- The driving time to a number of orders closest to the seed in the clusters. The number of orders is a fixed percentage of the total number of orders in a cluster.
- The driving time to a number of orders farthest away from the seed in the clusters. The number of orders is a fixed percentage of the total number of orders in a cluster.

We may need to clarify how we determine a fixed percentage of the total number of orders in a cluster. When we have a cluster of ten orders and we consider 55% of the total number of orders in a cluster, we should evaluate 5.5 orders. That is not possible, so we always round up this number. In this example we need to round up to six orders.

We calculate the driving time with the method above. In Chapter 6, we validate these alternatives. The shift duration for each cluster is defined as in Equation 9.

$$\begin{aligned}\text{Shift duration} \\ &= \text{Maximum duration of the shift} \\ &- \text{ driving time} - \text{breaks} - \text{service time}\end{aligned} \quad \text{EQUATION 9}$$

5.1.2. INCENTIVES

In Section 5.2, we describe the working of our approaches. We suggest two approaches: a parallel approach and a sequential approach. For the understanding of this section, we need to introduce some common ground of these approaches. The parallel and the sequential approach base on the same concept. They both make clusters by adding an order (or a group of orders) to an existing cluster. In this section, we define two different incentives to add an order. Both approaches consist of multiple runs. In each run, we evaluate all created clusters and use the cluster with the highest incentive.

SMALLEST RADIUS

The main goal of our research is to generate clustered trips. We want to deliver all orders in the same city with as few trucks as possible. In Chapter 3, we introduced the circle covering method of Savelsbergh (1990). This method is based on the location of an order. We take an unplanned order and call this the candidate seed order. Savelsbergh extends the circle by successively adding the orders closest in distance to the candidate seed order to the circle. The method stops when adding an order to the circle makes the sum of the load of the orders in the circle larger than the vehicle capacity. These steps are repeated for all unplanned orders. For each candidate seed order, Savelsbergh calculates the radius of the accompanying circle. The radius is the distance in a straight line between the candidate seed order and the order in the cluster farthest away from the candidate seed order. Subsequently, the method sorts these circles on increasing radius and uses this information to create routes simultaneously.

We use the circle covering method to define the first incentive. The circle covering method chooses the candidate seed order with the circle with the smallest radius. Consequently, the method attempts to keep the trips as clustered as possible. We concluded in Chapter 4 that the extent to which the routes were clustered differs in the current situation and the desired situation. A wish of the planners is to minimize the number of trucks that visit a city. That is not guaranteed with this method, but the chance increases. If there are multiple orders in a city, in general these orders are closer to each other than to an order outside the city. We verify this with dataset of Zeeman. We find that in 75% of the cases, an order in the same city is closest to the order. Even if the other order in the city is not the order closest to the seed order, we find that in 99% of the orders that an order in the same city is always in the top two of orders closest an order in the same city. That guarantees that the orders are covered by the circle for that seed order. However, there are only a few cities which have enough orders to fill the full capacity of a truck. The radius of the orders in the other cities depends on the density of the orders in the region of the seed order. However, when the density round a city is low, the chance that the orders in that city are delivered by different trucks is low. Orders in a city with multiple orders or in the neighborhood of such a city have a higher incidence to be chosen as seed order.

We call a cluster a set of orders that is added to a circle. Planners base their planning partly on visual attractiveness. The conclusion of the planners whether or not a planning is clustered enough is partly based on the visual attractiveness for the planners. We can use distance based on a straight line, or other criteria such as driving time or driving distance. Here, we choose for the distance in a straight line between the seed order and the order farthest away from this seed order as the radius of that cluster, because this fits better to the intuition of the planners. An order may be closer to the seed order in driving time, but geographically further away. That makes another criteria harder to explain to outsiders.

The cluster with the highest incentive is the cluster with the smallest radius. The exact way a cluster is generated differs between the parallel and the sequential approach. However, both approaches use the same general steps and logic. Figure 27 depicts the process of selecting the cluster with the smallest radius. The point of departure for each approach is a set of clusters. Each cluster consists of one or multiple orders and has a candidate seed order. First, we extend the cluster by merging the cluster with the order that increases the radius of the cluster the least to the current cluster, while remaining the cluster feasible of the merging option. We call this merged cluster the merging option for that cluster. We check the feasibility with the definitions of Section 5.1.1. For each merging option, we determine the radius. When we determined all merging options, we find the merging option with the smallest radius. The next steps depend on the chosen approach. We discuss those steps in Section 5.3 for the parallel approach and in Section 5.4 for the sequential approach.

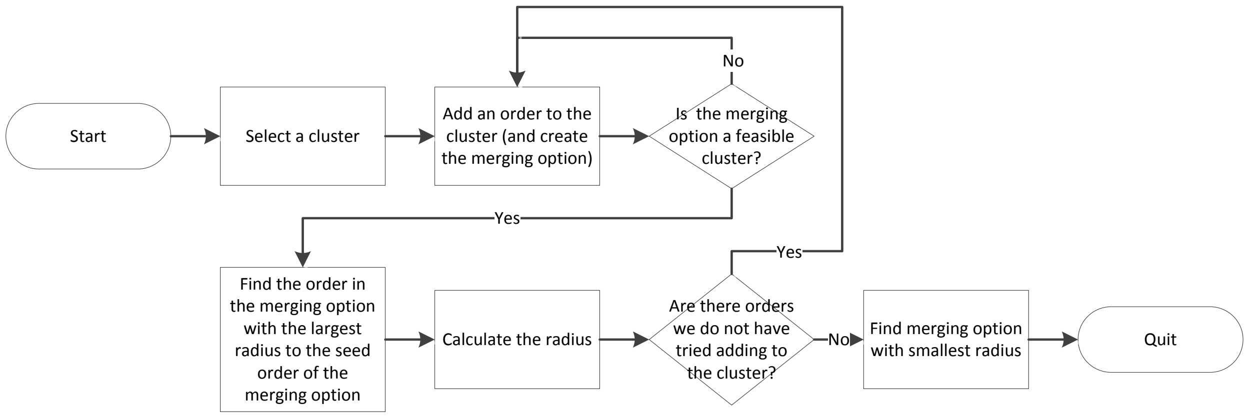


FIGURE 27 - FLOWCHART FOR FINDING THE SMALLEST RADIUS

LARGEST DIFFERENCE BETWEEN RADII

One of the main causes of our problem is that the choice for a shift influences the planning. Recall the example of a city with five orders of which two have a time window. The current algorithm starts with a shift in which it is only possible to deliver three of the five orders. An additional truck is sent to the city to deliver the other two orders. However, it is possible to deliver all five orders in the same shift (see Chapter 1 for a detailed description of this example). This example shows the importance of the right shift choice.

In the literature, we find multiple articles that use the regret factor to determine which order to add to a cluster or a route. The regret factor shows the difference in an incentive if we merge Cluster A with Cluster B, instead of Cluster A with Cluster C. We want to cluster the orders which locations are close to each other in distance. However, due to the time windows not all orders can be in the same cluster. The time windows of the orders reduce the number of feasible options for merging a cluster with an order. The majority of the orders can be delivered in at least two different shifts. However, most clusters can only be delivered in one shift, since that shift is the only overlapping time window of all orders in the cluster. In the planning of Zeeman, we have only limited vehicles available for each shift. We explain the regret with an example. Figure 28 and Figure 29 show two different situations. In both figures, the smallest circle represents the cluster if we use Shift I and the largest circle the cluster if we use Shift II. In Figure 28, the difference between the radii of the two clusters is small and we probably will have not much regret if we use Shift II instead of Shift I. In Figure 29, the difference between the radii of the two clusters is large. We probably regret it, if we do not use Shift I.

The point of departure for both the parallel and the sequential approach is a set of clusters. Each cluster consists of one or multiple orders and has a candidate seed order. We use the method of the smallest radius to extend the clusters. In the smallest radius method, adding orders determines which shifts are feasible for this cluster; the overlapping delivery time of all orders in the cluster is reduced. In the largest difference method, we set a shift before we start merging the cluster with orders. For each cluster, we determine the merging option with the smallest radius if we use Shift I.

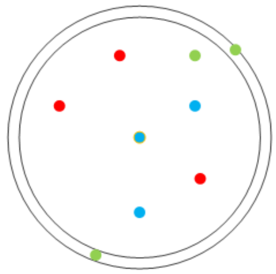


FIGURE 28 - SMALL DIFFERENCE BETWEEN RADII

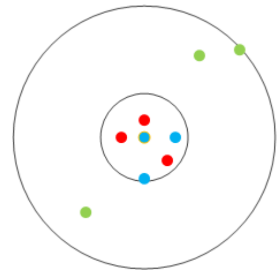


FIGURE 29 - LARGE DIFFERENCE BETWEEN RADII

We repeat this step for all available shifts. We now calculate the regret if we use a shift instead of another available shift. Since we want a solution that is as clustered as possible, the only relevant difference is that between the radius of the merging option for the cluster with the smallest radius and the radius of the merging option for the cluster with the second smallest radius (Equation 10). Figure 30 shows the flowchart of this method.

$$\begin{aligned}\text{Difference}_{\text{candidate,shift}} \\ &= \text{second smallest radius}_{\text{candidate,shift}} \\ &- \text{smallest radius}_{\text{candidate,shift}}\end{aligned} \qquad \textbf{EQUATION 10}$$

If it is only possible to deliver the merging options of a candidate seed order with one shift, we set the difference of the cluster very large and subtract that number with the smallest radius of the shift the cluster can be delivered with. In such way, we guarantee that the cluster is delivered with the required shift. But since we subtract the radius, we choose the cluster with the most clustered merging option first.

The largest difference method stops when we evaluated all clusters. We now have the difference between the shift with the smallest radius and the shift with the second smallest radius for each candidate seed order. We determine the seed order with the largest difference. The merging option with the smallest radius for that candidate seed order is set as a new cluster. The next steps depend on the chosen approach. We discuss those steps in Section 5.3 for the parallel approach and in Section 5.4 for the sequential approach.

5.2. COMBINING CHOICES

In Section 5.1, we discussed the options we have within an approach to make the plan more clustered. However, we did not define the framework in which we are going to use these choices. In Chapter 3, we introduced some algorithms we found in literature that deal with the problem of making a clustered solution in plans with time windows. We concluded that the most promising approaches are a parallel approach and extensions of the sequential insertion algorithm. The latter we call the sequential approach. The parallel and the sequential approach have overlap on some points. Therefore, we explain the general steps of these two approaches and we contrast these approaches in Section 5.2.1. In Section 5.3 and 5.4, the concepts of Section 5.1 are adapted and specified where needed. We discuss the alternatives of these approaches in Section 5.2.2.

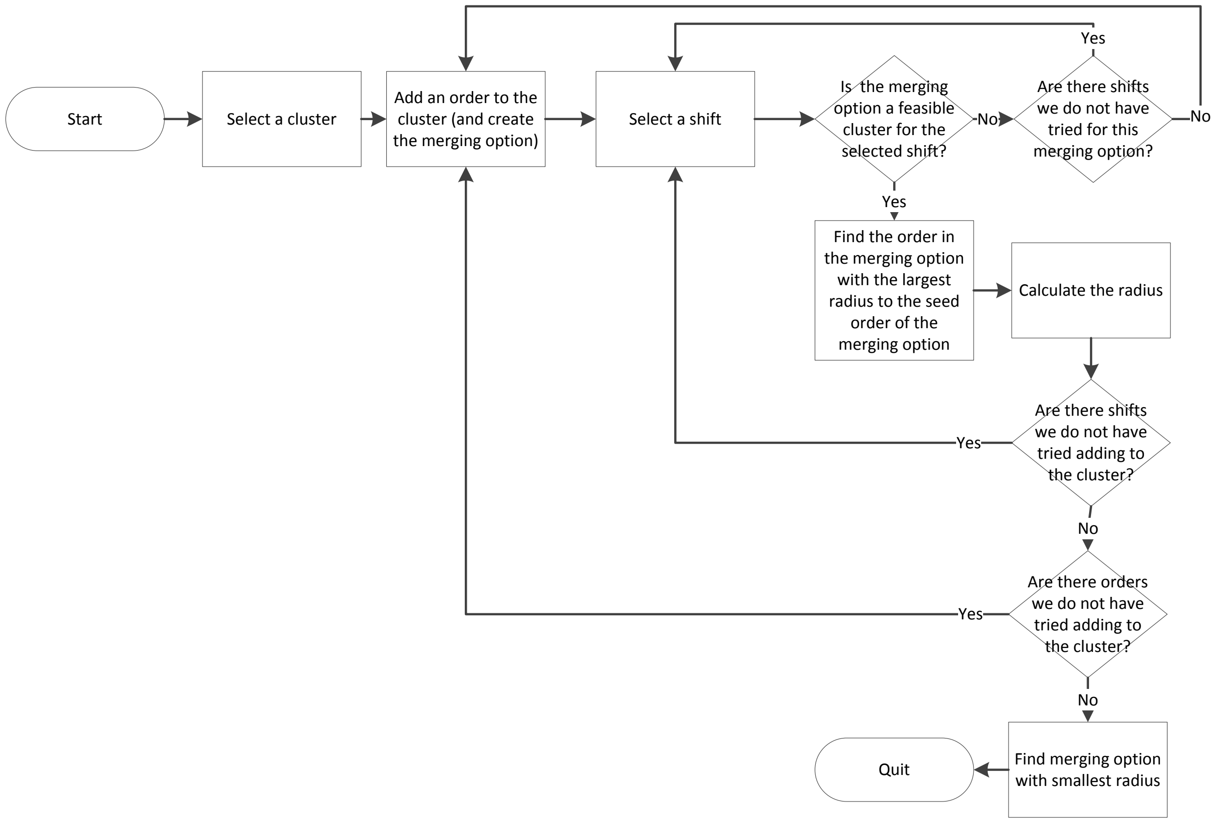


FIGURE 30 - FLOWCHART LARGEST DIFFERENCE BETWEEN SHIFTS

5.2.1. THE PARALLEL AND THE SEQUENTIAL APPROACH

The two most promising approaches we found Chapter 3 are a parallel approach and a sequential approach. Both are still a broad term for a set of approaches. In this section, we describe and contrast the approaches in the way we use them in this research.

PARALLEL APPROACH

In the parallel approach, we simultaneously generate the clusters. The general idea behind the approach is that we create clusters by adding a cluster to another cluster. We repeat this step until it is no longer possible to merge clusters without violating restrictions. The orders that are in a cluster at the end of this step are delivered in one trip.

We start the parallel approach with a separate cluster for each (unplanned) order. This order is the only order in the cluster. We create a new cluster by merging two existing clusters. To find the best option, we define and evaluate all merging options. A merging option is a combination of two existing clusters. We can combine each cluster with all other existing clusters at this moment; if n is the total number of clusters, we can combine this cluster with n-1 other clusters. This makes there are $\frac{n-1}{2} \cdot n$ merging options per run. We choose the merging option that is best according to the incentive defined in Section 5.1. We stop when it is no longer possible to merge clusters without violating a restriction.

In the parallel approach, each cluster contains a set of orders that can be delivered in one trip. In literature, mainly two alternatives are used to determine whether we are satisfied with the current situation and stop searching for new clusters. In the first option, we determine the number of clusters we want to use. When we achieved this number of clusters, we stop the algorithm. Due to the time windows of the orders in our cases, it is not easy to determine the number of clusters we need. Therefore, we use the second option, namely reduce the number of clusters till no reduction is possible without violating any constraints. We defined these constraints in Section 5.2.1.

Sequential approach

A sequential approach considers the clusters one by one. An advantage of the sequential approach is that it is closer to the current solution in TRP. This fits better in the wish of ORTEC to keep the changes to a minimum level. TRP currently uses the sequential insertion algorithm to generate the initial solution. Figure 31 recalls the four general steps of that algorithm.

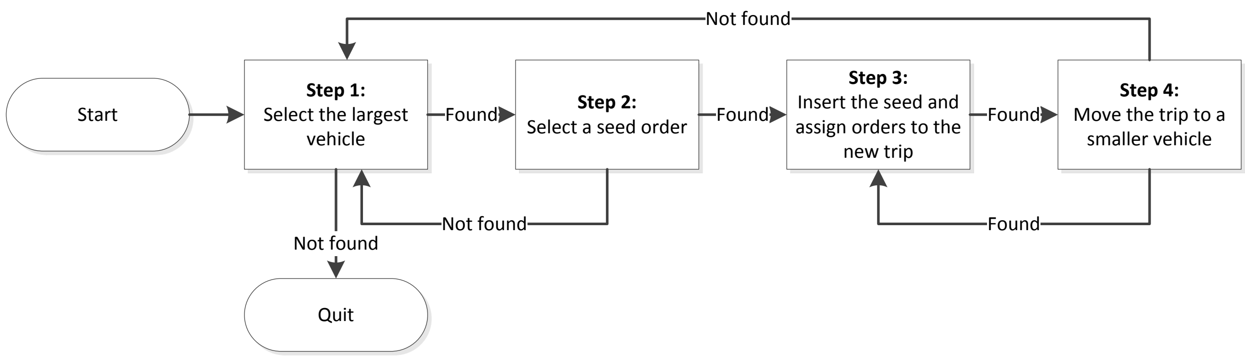


Figure 31 - General steps of the sequential insertion algorithm

In the sequential approach, we focus on adjustments to the first two steps of the sequential insertion algorithm. The criteria of the vehicle selection (Step 1) are accurate. However, we also established in the data analysis that the choice of a shift influences the seed choice (Step 2). Therefore, we consider an alternative of the sequential insertion algorithm in which we switch Step 1 and Step 2. Figure 32 shows the general steps of this variant of the sequential insertion algorithm. Furthermore, we use an alternative method to select the seed order. We do this by generating clusters with the incentives described in Section 5.1.2. We select the cluster with the highest incentive, and take the seed order of this cluster. In Chapter 4, we established that the assignment method (Step 3) is not the cause of the problem. This not means that adjustments to this step do not lead to a better solution. The last step of the algorithm, move the trip to a smaller vehicle, has no influence on the orders that are in a trip, and is therefore not relevant for this research.

Comparison

The parallel and the sequential approach differ in the way they create the clusters. We contrast the approaches by an example. In Figure 33 and Figure 34, we have the same set of orders. In Figure 33, we show the different steps in creating a cluster while using a sequential approach to generate the clusters. We find one compact cluster. However, to deliver the remaining orders, we have to make the second cluster large. In Figure 34, we use a parallel approach. We find that both clusters are relatively compact. Furthermore, the parallel approach has the advantage that there is a larger

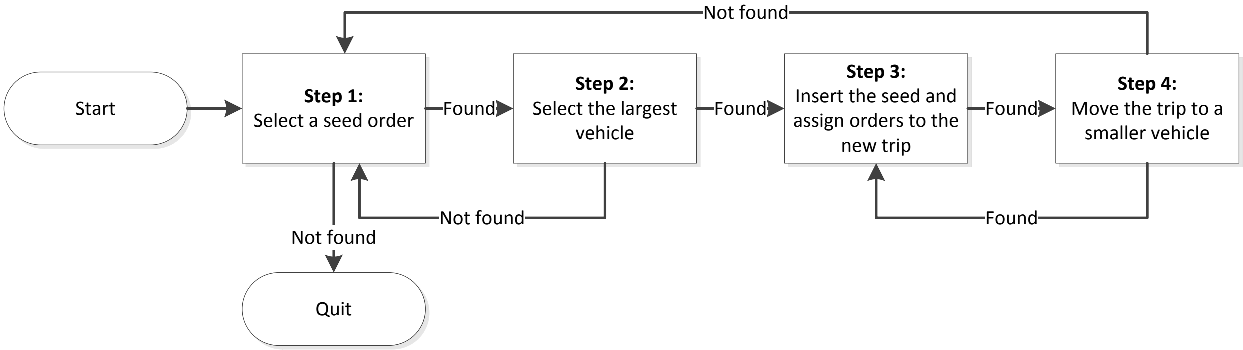


FIGURE 32 - GENERAL STEPS OF THE SEQUENTIAL INSERTION ALGORITHM WITH ADJUSTED SEQUENCE

probability that orders in the same city are kept together. If the two orders in the example just above the right corner are in one city, they are delivered in two different trips in the sequential approach, but in one trip in the parallel approach.

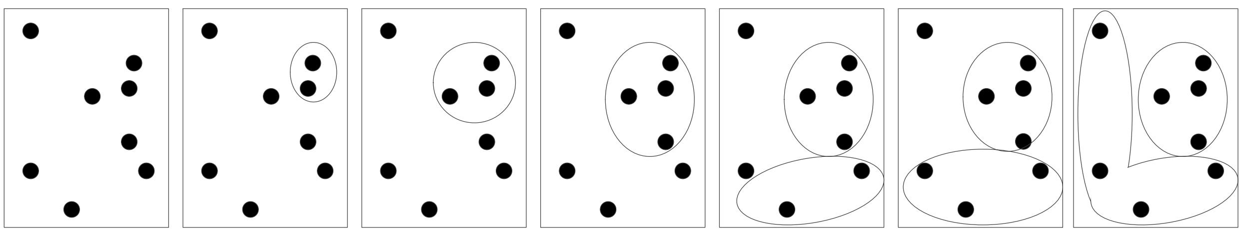


FIGURE 33 - CLUSTERS IN SEQUENTIAL APPROACH

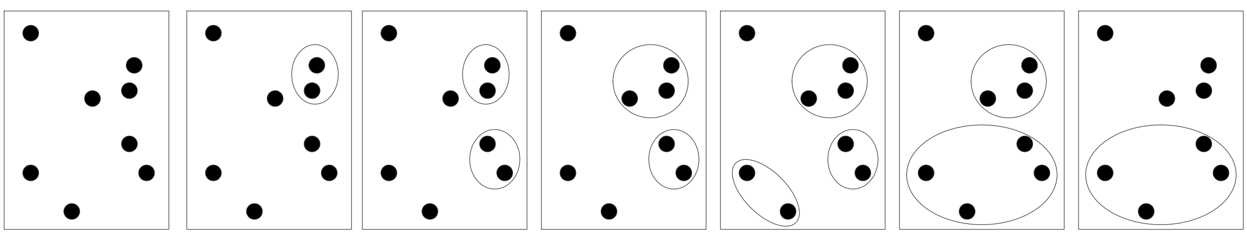


FIGURE 34 - CLUSTERS IN PARALLEL APPROACH

Note that the clusters in the parallel approach need to be unique; an order can only be assigned to one cluster. In the sequential approach, it is possible to assign the same order to multiple clusters, since we only choose one cluster at the end of the run, assign orders to that cluster, and then recalculate all clusters. In the parallel approach, we elaborate on the clusters of the previous run.

5.2.2. DISMISSED APPROACHES

In Chapter 3, we searched in literature for algorithms that focus on clustering plans with time windows. However, not all these algorithms are applicable for the situation in this research.

PARALLEL APPROACH

In Section 5.2.1, we described an alternative of the parallel approach. In that approach, we start with each unplanned order in a separate cluster and reduce the number of clusters by merging those clusters. We described the incentive we use in Section 5.1.2. However, there are also other incentives possible.

The savings algorithm is an example of an algorithm with a parallel approach. We described the savings algorithm in Section 2.3.2. This algorithm is already programmed in TRP. We perform a test in which we compared a planning generated with the sequential insertion algorithm with a planning generated with the savings algorithm. The performance indicators of this test are in Appendix D. We use a dataset of Zeeman to compare the savings algorithm with the sequential insertion algorithm. We compare the plans on the performance indicators defined in Chapter 4. The savings algorithm is not able to plan all orders in the initial solution. Although the costs, number of kilometers, and driving time all decrease, the algorithm scores worsen on clustering.

Instead of starting the approach with a separate cluster for each unplanned order, it is also possible to successively determine the number of clusters that we need and the orders we assign to a cluster. A well-known method in literature to determine the number of clusters to start with, is to determine the total load of all orders and divide that by the capacity of a truck. Some methods determine a seed order and assign orders to the route according to the location of an order in relation to the seed order. Other methods make clusters and subsequently route the orders in this cluster. Most sweep heuristics use the latter method. Such a method is hard to use in combination with time windows, since we cannot generate initial clusters solely based on the load of the orders. The best alternative in such a case is to generate an initial solution from which we divide the number of seed orders. That is a three phase approach. We consider that option in a later paragraph.

ADJUSTMENTS TO THE SEQUENTIAL INSERTION ALGORITHM

The sequential insertion algorithm that is currently implemented in TRP selects the most difficult customer as seed order (see also Chapter 2). In TRP, most difficult is translated to the order farthest away from the depot. However, also other characteristics of an order determine whether or not an order is difficult. We evaluate four characteristics of an order:

- Load

  We express the load of the order in the number of trolleys that are planned to be delivered at the location of the order. Each truck has a maximum capacity. In general, this makes it harder to add an order with a large load to a route, since there is a higher chance that the load of the order is larger than the remaining capacity of the truck. If we plan the order with the largest load first, this chance is smaller.

  We analyze the orders which are in the same city as another order. For our test, we use a dataset of Zeeman. The results of this test are in Appendix E. We conclude that most of these orders have an average load. Since we use other seed orders if we use the order with the largest load as seed, we get another solution. However, there is no indication that this solution will be more clustered. Therefore, we do not consider the order with the largest load as most difficult customer.

- Required vehicle type

  Some orders need to be delivered with a specific vehicle type. Other orders can be delivered with two or more vehicle types. However, there are only a finite number of vehicle types. For example, in the case of Zeeman we only have three vehicle types. When we use the required vehicle type as seed selection criterion, we only reduce the number of possible seed orders. We always need a second criterion to determine the seed order.

The vehicle type is linked to the geographical location of the order. When we assign vehicle types to the orders, we basically generate large zones. These zones may overlap. However, we still determine the seed based on the distance from the depot, similar to the selection criterion 'farthest away from the depot'. Therefore, we do not test the required vehicle type as seed selection criterion.

- Due date
  Each order has a latest delivery time, called the due date. In some plans, it is useful to plan the orders with the earliest due date first. In other plans, it more convenient to the plan the orders with the latest due date first. However, the cases we analyze do not have a significant difference in the distribution of the due dates. For that reason, we do not consider the earliest or latest due date as seed selection criterion.

- Length of the time window
  All orders in the plan have a time window (Chapter 2). The time window of some orders covers multiple days, of others only multiple hours. It is harder to plan orders with a small time window into a route, simply because there are fewer options to plan this order.

  We may choose the order with the smallest time window as seed order. However, in this approach we face a problem. There are orders that have time windows with the same duration. We need a second criterion to distinguish which of these orders should be planned first. From the options we discussed in this section, the required vehicle type and the orders farthest away from the depot are the best alternatives. Again, the required vehicle type only limits the number of options. Also the criterion farthest away from the depot only limits the number of options. However, the chance that two orders have the same distance to the depot is much smaller than the chance that two orders have the same required vehicle. Therefore, we choose for farthest away from the depot as second criterion.

  We select the order that satisfies all statements below:
  a. The order is not assigned to a vehicle yet
  b. The order has the smallest time window among all unassigned orders
  c. The order is the order farthest away of the depot among all unassigned orders with that (smallest) time window.
  d. The order can be delivered during the shift time of the vehicle
  e. The order amount is lower than the capacity of the truck

  We perform a preliminary test to get an indication of the quality of such an approach. The results of this test are in Appendix F. The approach was not able to plan 34 orders. Therefore, it is not fair to make a comparison with the performance indicators of the plan made by TRP. The manual adjustments that are needed to plan the remaining orders will change the performance indicators. Due to the large number of unplanned orders, we do not consider this approach as possible solution.

AN SOLUTION APPROACH THAT SELECTS ORDERS BASED ON THEIR GEOGRAPHICAL LOCATION

Many solution approaches that focus on clustering use the geographical location of an order to determine the clusters. These are solution approaches such as the sweep algorithm, the general assignment method, and derivations of these algorithms. Especially the general assignment method has a high incidence in literature and gives good results with respect to clustering. However, in Chapter 3, we concluded that it is hard to consider time windows in these kinds of algorithms. In literature, we did not find a suitable alternative that matches with our solution requirements.

If we make a separate set of orders for each shift type, it is possible to use an indicator. The only available orders are those orders that fit in the shift. Therefore, we have to bother less about the time windows. However, we need to define the shift before we start the algorithm. We test this approach on a dataset of Zeeman and compare the results (Appendix G) with the plan we generate with the algorithm that is currently used in TRP. We find that the costs are reduced, but that the plan is less clustered and the visual attractiveness is not very good in comparison to the plan generated by TRP. These results are not promising enough to consider further investigation in this research.

THREE PHASE APPROACH

The TRP's current algorithm and all solution approaches we mentioned in this chapter till now consist of two phases: an initial solution and improvement steps. In Chapter 3, we mentioned some algorithms that consist of three phases. The first phase is an initial phase where a first solution is generated. From this solution the seed orders are derived. The remaining of the solution is thrown away. In the second phase, a new initial solution is generated, with the seed orders as point of departure. The third phase is the improvement steps. The outcomes of these algorithms are relatively good. However, the calculation time of such an approach is higher, since we generate an initial solution twice.

To make the algorithm successful, it is important to choose the seed orders at the end of the first phase wisely. Earlier in this section, we already explored the possibilities of a good seed selection method. Due to time limitations for this research, we cannot explore both options intensively. A three phase approach changes the way of working compared to the current algorithm. This, in addition to the large calculation time required for the first and second phase, makes us conclude not to investigate this option.

5.3. PARALLEL APPROACH

In Section 5.2, we introduced the general idea behind the parallel approach. In Section 5.1, we introduced the feasibility check and the incentives we use. In this section, we adapt these concepts and specified where needed.

5.3.1. ASSIGN ORDERS TO CLUSTERS

In the first step of the parallel approach, we create the clusters. In Section 5.2, we explained that we start with the situation with each order in a separate cluster. We want to merge the two clusters with the highest incentive. In Section 5.1.2, we described how we find the cluster with the highest incentive. Note that for the parallel approach, we do not add a single order to an existing cluster, but we add another cluster. This has no influence on the working of the approach. When we found the best merging option, we set the merging option as new cluster. With this new set of clusters, we repeat the steps above. We stop, if we cannot find a merging option without violating any of the restrictions. Figure 35 shows a flowchart of these steps.

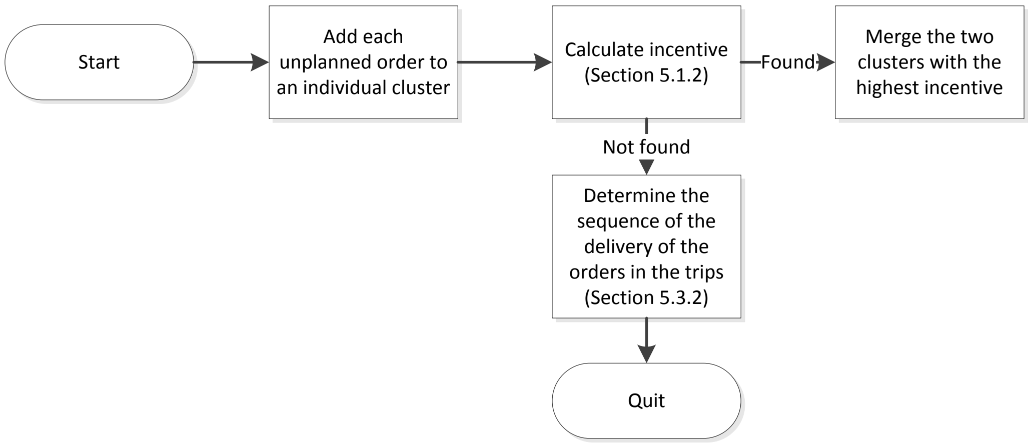


FIGURE 35 – STEPS IN THE PARALLEL APPROACH

We explain the parallel approach with an example. We have six orders, order A to F. Table 8 gives the starting point of the parallel approach; all orders are in a separate cluster. To decide which two clusters are the best clusters to merge, we need to evaluate all merging options. Table 9 shows the merging options for the first cluster which contains only order A. For each merging option we check the feasibility. We also evaluate the merging options for all other clusters. Merging Cluster 2 and 3 is the option with the highest incentive. We merge those two clusters. Table 10 shows the set of clusters at the beginning of the second run. Again, we evaluate all merging options and choose the best option. Table 11 shows the merging options for the first cluster in the second run. We repeat those steps, until no feasible merging options are left. Table 12 shows the final clusters of this example.

TABLE 8 – STARTING POINT OF THE FIRST RUN

Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6
Order A	Order B	Order C	Order D	Order E	Order F

TABLE 9 - MERGING OPTIONS FOR CLUSTER 1 IN THE FIRST RUN

Merging option 1	Merging option 2	Merging option 3	Merging option 4	Merging option 5
Order A	Order A	Order A	Order A	Order A
Order B	Order C	Order D	Order E	Order F

TABLE 10 – STARTING POINT OF THE SECOND RUN

Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Order A	Order B	Order D	Order E	Order F
	Order C			

TABLE 11 - MERGING OPTIONS FOR CLUSTER 1 IN SECOND RUN

Merging option 1	Merging option 2	Merging option 3	Merging option 4
Order A	Order A	Order A	Order A
Order B	Order D	Order E	Order F
Order C			

TABLE 12 - FINAL CLUSTERS

Cluster 1	Cluster 2
Order D	Order B
Order E	Order C
	Order A
	Order F

5.3.2. GENERATE ROUTES

The last step in the parallel approach is to determine the sequence in which the orders are delivered. We use the exact composition of the clusters we generated in the previous step of the parallel approach. Therefore, we only need to determine in which sequence we execute the route. By doing this, we reduce the problem to a travelling sales man problem with time windows. How to solve the travelling sales man is outside the scope of this thesis. In our test, we use the logic for generating routes already implemented in TRP. We described this logic in Chapter 2.

5.4. SEQUENTIAL APPROACH

In the sequential approach, we consider two possibilities. In the first possibility, we successively select a vehicle, select a seed order, assign orders to the trip, and move the trip to a smaller vehicle. In the second possibility, we switch the first and the second step. An advantage of this second possibility is that we do not limit our self by the need to find a seed order that can be delivered with the shift of the chosen vehicle.

In Section 5.2, we introduced the general idea behind the sequential approach. In Section 5.1, we introduced the feasibility check and the incentives we use. In Section 5.4.1, we describe an additional incentive, specifically for the sequential approach. We adapt the concepts of Section 5.1 in Section 5.4.2 and specify them where needed. We describe the approach of the first possibility in Section 5.4.3 and the second possibility in Section 5.4.4.

5.4.1. ADDITIONAL INCENTIVE FOR THE SEQUENTIAL APPROACH

As we explained with the example in Section 5.2.1, the clusters in the sequential approach are constructed in a different way than in the parallel approach. There is a higher chance that orders become more isolated, which leads to clusters that cover a large distance. When the depot is chosen strategically, it is assumed that the density of the orders in the neighborhood of the location of the depot is higher than the density of the orders further away from the depot. For the case of Zeeman, this is certainly true. With the incentives we defined in Section 5.1.2, this means that we first generate orders in the neighborhood of the depot and slowly expand to orders further away from the depot. In general, the orders farther away from the depot a harder to plan, so it is preferred to start with these orders.

In Chapter 3, we introduced some methods that divide the map of the orders into zones or rings. In our approach, we also divide the map into multiple rings. The rings are based on the distance in a straight line from the depot. In literature, there are multiple theories available that determine the optimal size of these zones. However, in most cases these calculations are complex. Due to the time available for this research, we have to choose to base the radius of our zones on an educated guess. We set the radius on 25 kilometers.

The most difficult customers are in general in the most outer circle. Therefore, we add an additional criterion to the seed selection step: the seed order should be in the outer ring where there are unplanned orders.

5.4.2. Seed selection procedure and assigning orders to trips

We select a seed order by generating clusters and choose the seed order accompanying the cluster with the highest incentive. In each run, we set all unplanned orders at the beginning of a run to candidate seed orders. For each candidate seed order, we determine the accompanying cluster with the method explained in Section 5.1.2. Next, we choose the cluster with the highest incentive. We retrieve the candidate seed order of this cluster. We use that seed order in the next step of the approach, assigning orders to the trip. With the cheapest insertion method, we assign orders to trip. We repeat these steps until all orders are assigned to a cluster. Figure 36 gives the flowchart for this procedure.

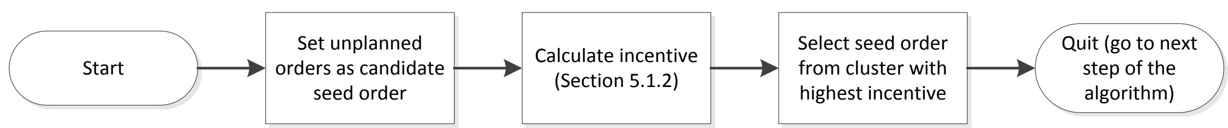


Figure 36 - Flowchart for using clusters as seed selection method in the sequential approach

We explain this part of the sequential approach with an example. We have six orders; Order A to F. Table 13 gives the starting point of the approach. We first select Order A as candidate seed order. We search for the order that increases the incentive of the candidate cluster, consisting of the seed order and one other order, the least. This turns out to be Order F. We now have a candidate cluster consisting of Order A and F. We search for an additional order we want to add to the candidate cluster and find Order B. We extend the candidate cluster that was already consisting of Order A and F with Order B. Again, we search for the order that increases the incentive of the candidate cluster the least. We find Order C. However, a candidate cluster of Order A, B, C, and F is not a feasible cluster according to the criteria defined in Section 5.1.1. We now stop[4] and repeat those steps for the next candidate seed order. All candidate clusters are given in Table 14. Note that candidate clusters I, II, and IV and II and IV consist the same orders. However, they all have a different candidate seed order and therefore a different radius (the row with the numbers in Table 13 and Table 16). Say that the candidate cluster with the highest incentive is candidate cluster II. The seed order of candidate cluster II was Order B. In the assignment step of the approach, Order A and F are assigned to the trip. The unplanned orders at the beginning of the second run are Order C, D, and E (Table 15). We again determine all candidate clusters. Candidate cluster A turns out to be

[4] It may be that adding a neighboring order also leads to a feasible result. Later in this section, we explain how we search for alternative orders to add to the cluster.

the cluster with the highest incentive. In the assignment step of the approach, Order D is assigned to the trip. The only unplanned order is Order E. We choose this order as seed order, but have no orders remaining to assign to this order. This order is alone in a trip. Table 17 gives the final clusters for this example.

TABLE 13 – UNPLANNED ORDERS BEFORE THE FIRST RUN

Order A	Order B	Order C	Order D	Order E	Order F

TABLE 14 – CANDIDATE CLUSTERS AFTER FIRST RUN

Candidate cluster I	Candidate cluster II	Candidate cluster III	Candidate cluster IV	Candidate cluster V	Candidate cluster VI
0.79	0.67	0.88	0.91	0.75	0.81
Order A	Order B	Order C	Order D	Order E	Order F
Order B	Order A	Order D	Order C	Order D	Order A
Order F	Order F			Order F	Order B

TABLE 15 – UNPLANNED ORDERS BEFORE THE SECOND RUN

Order C	Order D	Order E

TABLE 16 – CANDIDATE CLUSTERS AFTER FIRST RUN

Candidate cluster I	Candidate cluster II	Candidate cluster III
0.88	0.91	0.95
Order C	Order D	Order E
Order D	Order C	Order D

TABLE 17 – FINAL CLUSTERS

Order B	Order C	Order E
Order A	Order D	
Order F		

Note that it is not necessary that the same orders that are in the candidate cluster of the seed order are assigned to the trip. Due to a different assessment in the assignment of orders to cluster and assignment of order to trips, the outcome may differ. When we generate the clusters, we stop assigning orders to the cluster as soon as the order which increases the incentive of the cluster the least does not result in a feasible cluster. However, there may be an order which increases the incentive only a little bit more, and is therefore a good alternative. This order is not considered. Figure 37 depicts this situation. We choose Order A, where Order B is also a good option. In Chapter 6, we validate whether it gives a significant difference if we consider other options in the neighborhood of the cluster. An advantage of not using the clusters in the way we assigned them, is that the risk we mentioned in Section 5.1.1 about not considering the sequence of the orders in the trip is no longer an issue. In the assignment method of the trips, we consider the sequence and recheck the feasibility. TRP only assigns an order when this leads to a feasible cluster. Of course, the more reliable the clusters are, the more accurate the seed choice is.

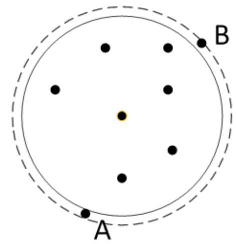


FIGURE 37 - A SITUATION IN WHICH ANOTHER ORDER IS GIVES A GOOD ALTERNATIVE

5.4.3. FIRST VEHICLE SELECTION, THEN SEED SELECTION

The sequential insertion algorithm successively selects the vehicle and the seed order. When we know the vehicle, we know the shift in which the trip is executed. This considerably reduces the number of orders we need to evaluate, since we can skip all orders that are not feasible to be delivered in this shift.

We need to make some special remarks when we use the largest difference incentive. Although we already know in which shift the trip will be executed, we still need to calculate a cluster for each possible shift for each candidate seed order. This information is necessary to calculate the difference between the radii of the shifts for each candidate seed order. We add one additional requirement to the selection process of the best option compared to the method described in Section 5.1.2. We only consider the candidate seed orders of which the shift with the smallest radius is similar to the shift of the chosen vehicle. From that set of candidate seed orders, we choose the candidate seed order with the largest difference between the radii of the shift.

5.4.4. FIRST SEED SELECTION, THEN VEHICLE SELECTION

In Chapter 4, we established that the choice for a shift influences the composition of the orders in a trip. When we first select a vehicle and the select a seed, there is no guarantee that we cannot find another order in another shift which has a higher incentive. This may lead to less preferred situations later in the planning process. Figure 38 shows a situation in which two candidate clusters are overlapping. The large blue circle is the cluster with the highest incentive for the selected vehicle. The smaller grey circle is a candidate cluster for another vehicle. If we plan the cluster of the large blue circle first, we need to find a new cluster for the grey circle, since the overlapping order is already planned in another (the blue) cluster. The new incentive for the grey cluster is probably lower than the incentive of the current cluster. If we use the largest difference as incentive, we expect that the effect on the planning is smaller, since we at least consider the different shifts of the seed order and thereby get some more information about the neighborhood.

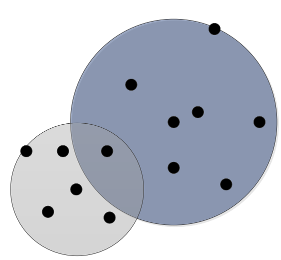


FIGURE 38 - OVERLAPPING CLUSTERS IN DIFFERENT SHIFTS

When we select the seed order in the first step of the approach, there is no restriction on the vehicle, and thereby on the shift. After we determined the seed order accompanying the cluster with the highest incentive, we determine the vehicle type of the trip and the shift. We can retrieve this information, since we defined for each order the required vehicle type and the time windows, and we check these requirements with the feasibility check. In most cases, only one vehicle type and shift combination remains. However, if more options are possible, we fall back on the original four selection criteria of the vehicle selection in the sequential insertion algorithm. To define a complete vehicle selection procedure for this approach, we add one additional statement to the original procedure. All statements the vehicle should fulfill are:

a. Select the vehicle with the shift that is determined in the seed selection step
b. Select the vehicle with the highest vehicle priority that is empty
c. Select the largest empty vehicle
d. Select the vehicle with the lowest identification number

After we choose a vehicle, we go to Step 3 of the sequential insertion algorithm, assign orders to the trip.

5.5. Conclusion

In this chapter, we answer the question "*How should an approach for the clustering of orders in a vehicle routing problem with time windows look like?*". We answered this question by adjusting the most promising approaches from the literature review: a parallel approach and an adjustment to the sequential insertion algorithm (called the sequential approach in this thesis).

Both approaches are based on generating clusters. We first defined a feasibility check for these clusters. A cluster is feasible when:

1. The total load in the vehicle is lower than the capacity of the truck
2. All orders in the clusters can be delivered within their time window
3. The total duration of the shift is shorter than the maximum duration of that shift

The capacity violation is not hard to check. The second constraint, the time windows of the orders, is more difficult. Especially since the position of the order in the trip is not known. Therefore, we do not know the exact delivery time of the order. We do check whether it is possible to deliver an order within a shift, considering the driving time from the depot to the order, the service time of the order, and the departure and arrival time of the vehicle at the depot. The third constraint, the duration of the shift, is calculated by the sum of the driving time, waiting time, service time, and breaks in that shift. The service time is known when we know the orders in the cluster and the breaks are known if we know the shift type. We cannot calculate the waiting time, since we do not know the sequence in which the orders are delivered. For the same reason, we cannot calculate the driving time. However, since the total shift duration consists for a large part of driving time, we defined an estimate for the driving time.

We defined two incentives that are used to generate the clusters. The first incentive is the smallest radius. We generate a cluster by find the order (or group of orders) that increases the radius of the cluster the least, while remaining the cluster feasible. The steps are summarized in Figure 27. The second incentive is the largest difference between two shifts with the same candidate seed order. For each candidate seed order, we generate a cluster for each feasible shift. The incentive to generate the cluster is the smallest radius incentive. For each order, we subtract the radius accompanying the shift with the smallest radius for that seed order with the radius accompanying the shift with the second smallest radius for that seed order. The steps are summarized in Figure 30.

In the parallel approach, each order is in a separate cluster in the starting situation. We use the described incentives to simultaneously merge clusters to generate a solution. Figure 35 shows the steps of the parallel approach.

The sequential approach is an adjustment of the approach currently used in TRP, the sequential insertion algorithm. We developed two variants: in the first variant, the sequence of the steps is equal to the sequential insertion algorithm (Figure 31). In the second variant, we switched the first and the second steps, such that the sequence becomes as follows (Figure 32):

1. Select a seed order
2. Select a vehicle
3. Assign orders to the trip
4. Move trip to a smaller vehicle

An advantage of this variant is that the seed choice depends less on the vehicle choice.

In both variants, we determine the seed order by creating the clusters with the incentives. We select the seed order of the cluster with the highest incentive. In case we use the variant of the sequential approach in which the vehicle selection is the first step, we add an additional criteria for the seed selection, it should be possible to deliver the seed order in the shift of the selected vehicle. This reduces the number of possible seed orders. In case we use the variant of the sequential approach in which the vehicle selection is the second step, the vehicle should drive the shift required for the selected cluster. We did not change anything to the third and fourth step of the approach.

Summarizing, we developed six approaches in this chapter. Figure 39 depicts the approaches. We validate and compare these approaches in Chapter 6.

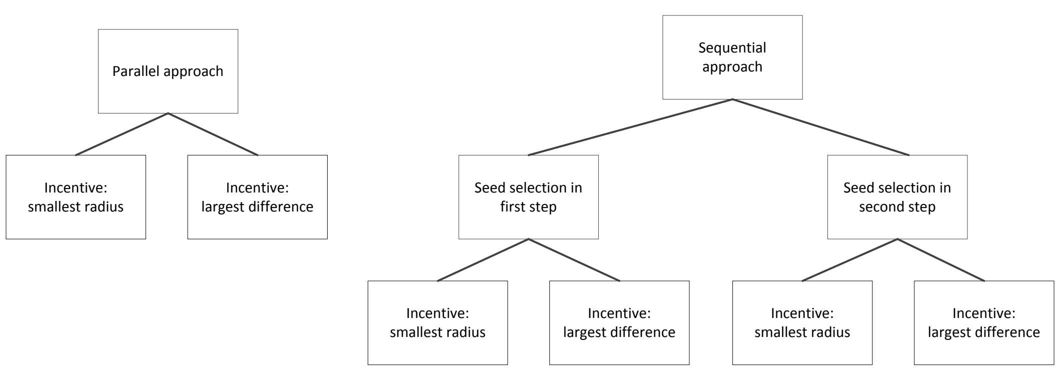


FIGURE 39 - APPROACHES

6. Validation and comparison

In Chapter 5, we proposed different approaches to improve the planning. In this chapter, we validate these approaches. Since clusters are the foundation of all approaches, we first validate these (Section 6.1). Subsequently, we test the different approaches defined in Chapter 5 and validate these (Section 6.2). In Section 6.3, we contrast the different approaches and incentives. The conclusion is in Section 6.4.

6.1. Validate clusters

The clusters play an important role in the planning. As explained in Section 5.1, we base the clusters on three aspects: the load of the orders, the time windows of the orders, and the duration of the trip. The input parameters for the first two aspects are fixed. For the latter, we discuss some alternatives. We validate these alternatives in Section 6.2.1. In Section 6.2.2, we discussed the accuracy of the clusters.

6.1.1. The duration of the trip

The duration of the trip is the sum of the driving time, service time, waiting time, and the breaks. In Chapter 5, we established that we can determine the service time since we know which orders are in the trip and the breaks since we know the shift type. We also established that it is hard to give an estimate of the waiting time since the sequence of the delivery of the orders is unknown. The same applies to the driving time. However, the waiting time is a small part of the shift duration and we aim to avoid waiting time. We also aim to minimize the driving time. However, the driving time is a significant part of the total shift duration. Therefore, we only consider estimating the driving time.

In Section 5.1, we proposed a method to calculate the driving time by using the driving time from the seed order to a number of orders in the cluster. In this section, we validate whether the estimated driving time is matched with the driving time in TRP. In the driving time in TRP, we consider the position of the order in the trip. Therefore, this driving time is a more accurate representation of reality. To test whether the proposed method to calculate the driving time is a valid method, we estimate the driving time for 168 clusters. The number of orders in the clusters and the radius of the clusters differ between the different clusters. For each cluster, we perform a test in which we vary the number of orders we consider from n = 0 (no orders, therefore no driving time) to n = 10 (we consider 10 orders). Furthermore, we execute this cluster in TRP and retrieve that driving time.

Table 18 – Average variance between trip durations in percentages, number of orders

Experiment	n = 0	n=1	n=2	n=3	n = 4	n = 5	n = 6	n = 10
Closest to the seed order	-42.5%	-37.3%	-29.0%	-19.1%	-6.2%	10.2%	26.3%	69.9%
Furthest away from the seed order	-42.5%	-16.1%	6.4%	25.5%	41.9%	55.8%	67.1%	86.6%
Random	-42.5%	-32.2%	-25.2%	-15.0%	-6.8%	-3.8%	2.1%	16.9%

Table 18 gives the average variance in percentage between the estimated shift duration with our method and the shift duration of the cluster in TRP. We conclude that there is a large variance between the estimated shift duration and the shift duration calculated in TRP. When we consider the deviation of each single cluster instead of the averages, we find that the estimation of the shift duration is underestimated in clusters with a lower number of orders than the average number of clusters, and overestimated in clusters with a higher number of orders than the average.

TABLE 19 - AVERAGE VARIANCE BETWEEN TRIP DURATIONS IN PERCENTAGES, PERCENTAGE OF ORDERS

Experiment	n =0%	n=10%	n = 20%	n=30%	n = 50%	n = 60%
Closest to the seed order	-42.5%	-40.9%	-34.3%	-27.3%	-8.5%	4.3%
Furthest away from the seed order	-42.5%	-16.3%	-1.2%	11.3%	67.1%	86.6%

Table 19 gives the results when we calculate an estimation of the shift duration based on a percentage of the total number of orders in a cluster. The variances are closer to zero, which means that the estimations are closer to the executing shift duration. We vary the values, such that the deviation becomes as close to zero as possible. We gain the best result with orders closest to the seed order when we use the 57% orders closed to the seed order (-0.1% deviation), and 21% when we use the orders farthest away from the seed order (1.0% deviation).

We compare the distribution of the variances of the clusters (Figure 40). Both methods to estimate the driving time that use a percentage of the total number of orders in a cluster have the most orders close to a deviation of 0%. The variant which uses a percentage of the total number of orders furthest away from the seed has slightly more clusters with driving time close to the shift duration in TRP.

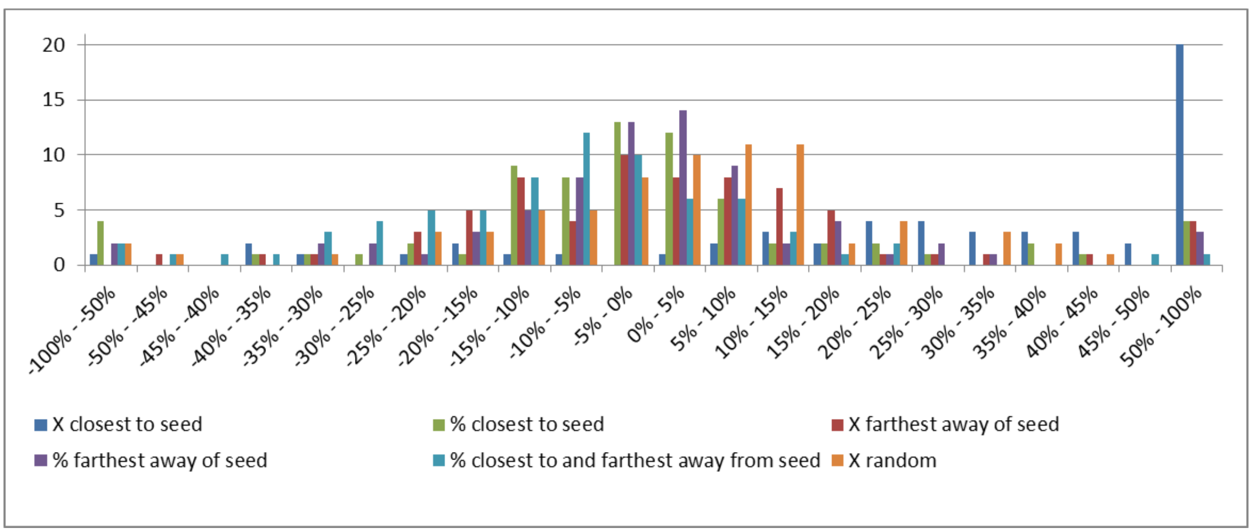


FIGURE 40 - DISTRIBUTION OF THE VARIANCE IN PERCENTAGES

We perform a statistical test to determine whether the differences between the estimated shift duration and the shift duration in TRP are similar. If we want to keep the confidence interval not larger than $\pm$ one hour, we need at least 119 observations. In our test, we had 168 observations. Therefore, we can conclude that when using a confidence interval of 95%, there is no significant difference between the estimated shift duration and the shift duration in TRP when we base the driving time on a percentage of the total number of orders. When we base the driving time on a fixed number of orders, the difference between the two shift durations is significant.

The difference between the shift durations is slightly less distributed when we use the 57% of the total number of orders farthest away from the seed order in a cluster, than we use the 21% of the total number of orders closest to the seed in a cluster. Therefore, we use the 57% orders farthest away from the seed order in our tests in this chapter.

6.1.2. Accuracy of the clusters

In Section 6.1.1, we focused on the trip duration. In this section, we focus on the accuracy of all aspects in a cluster.

Feasibility of the trips

In the remainder of this chapter, we use the term cluster to refer to the clusters generated with our approaches. We use the term trip for the clusters for which the sequence of the orders is determined.

In the parallel approach, we use the exact composition of the orders in the trips. Therefore, it is important that the clusters are feasible. In the sequential approach, this is less an issue, since we only use the seed order of the cluster calculated with the (parallel) approach. Of course, the clusters are feasible according to the criteria we defined in Section 5.1.1, but the assignment criteria in TRP are more extensive. We analyze the trips of the plan made with the parallel approach. We find that about 10.7% of the trips are not feasible. All of these trips have violations on the trip duration, or have orders in the trip that are not delivered in the time window of the order. In case the trip duration is exceed, this is caused by either a long driving time or waiting time between the orders. In about one third of the trips, both aspects are violated in a trip. In the remaining orders only one of the two aspects is violated. In about two third of the trips, the driver has to wait to deliver an order in the time window. In most trips, this waiting time is at least an hour. The waiting time extends the duration of the trip. The remaining shift duration is not long enough to deliver all remaining orders. Therefore, the vehicle is not back on the depot on time. In other trips, the remaining orders can be delivered in the remaining time span of the trip. However, the time windows of the orders are overlapping, such that it is not possible to deliver all orders in their time window.

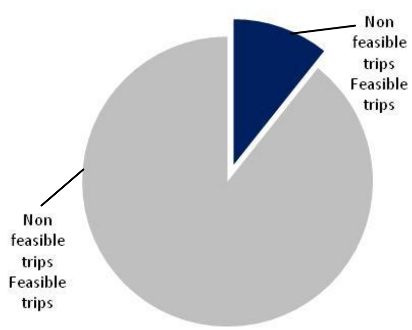


Figure 41 - Pie chart of feasibility of the trips

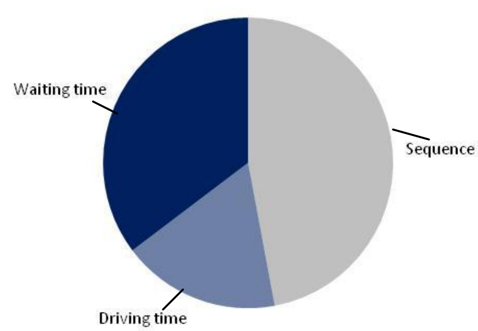


Figure 42 - Pie chart of different causes violations

In the sequential approach, we only use the seed order of the determined cluster. Therefore, the feasibility of the cluster is not equal to the feasibility of the trip. We find that all trips in the sequential approach are feasible. The orders are assigned to the trip with the assignment method of

TRP. Since this assignment is sequential, it is possible to consider the position of the order in the trip during the assignment.

SIMILARITY BETWEEN THE CLUSTERS AND THE TRIPS

In the parallel approach, the composition of the clusters is equal to the composition of the trips. Therefore, we focus in this section on the sequential approach.

The differences between the composition of the orders in the clusters determined in the approach and the executed trip are caused by the different assignment methods of the orders to the seed used by our approaches and the sequential insertion algorithm. In our approaches, we determine the clusters based on the smallest radius to the seed order. The assignment method in the sequential insertion algorithm searches for the order with the smallest insertion costs. This may be an order which has a larger radius to the seed, but is close to the order which is already assigned to the trip. The insertion costs would be larger if we need to drive to a new region for the order with the smallest radius.

The second cause of the difference between the clusters is related to the feasibility check (Section 5.1). In our approach, we stop assigning orders to a cluster when one of the defined constraints is violated. The assignment method of the sequential insertion algorithm searches for an alternative. We test whether the accuracy of the clusters improves if we search for alternative orders in the neighborhood of the location of the cluster. We consider searching for orders within a distance of 10% of the radius of the cluster and 20% of the radius. With a confidence interval of 95%, we need 81 observations to say something about the significance of the results. We test the alternative of 168 clusters. Test results show that the clusters generated with the approach with a search in 20% of the radius are slightly more representative for the clusters in TRP (an increase from 63% to 66%). However, statistical tests with a confidence interval of 95% show that this difference is not significant and cannot be assigned to the extension of the radius. Therefore, we do not search for orders in the neighborhood in the tests for our approaches. The main reason for the difference between the clusters in the sequential approach is the different assignment methods. In Section 7.4, we discuss some further research for this point.

6.2. TEST RESULTS

To validate the different approaches, we compare them on the performance indicators and on visual attractiveness. In Chapter 4, we defined performance indicators to compare the plan adjusted by the planners of Zeeman with the plan made by the current algorithm of TRP. In this chapter, we use the same performance indicators to determine the quality of our approaches. Table 21 gives the performance indicators for all tests. In Section 6.2.1 through 6.2.6, we discuss the most important and the most remarkable indicators for each approach. In Table 20, we declare the abbreviations we use in Table 21. More detailed numbers based on the different vehicle types can be found in Appendix H and I.

TABLE 20 - ABBREVIATIONS OF APPROACHES

Approach	Incentive	Abbreviation
Parallel	Smallest radius	P_SR
Parallel	Largest difference	P_LD
Sequential, vehicle first	Smallest radius	S_V1_SR
Sequential, vehicle first	Largest difference	S_V1_LD
Sequential, vehicle second	Smallest radius	S_V2_SR
Sequential, vehicle second	Largest difference	S_V2_LD

Furthermore, we evaluate the visual attractiveness of the plans. Although, this is a subjective metric, we try to examine this on two points:

- the extent of crisscrossing of trips over the map. We want to keep the region the trips visits as small as possible. For example, we want to avoid a trip that delivers orders in both Rotterdam and Arnhem. And,
- the extent in which we can distinguish the different trips. We want that trips cross each other as less as possible.

Figure 44 through Figure 49 depict the overview of The Netherlands, Belgium, Luxembourg, a part of France, and a part of Germany for each plan generated with one of the developed approaches. From these pictures, the difference will not always become very clear. However, the pictures give some idea about the visual attractiveness of the plans. For the analysis, we used the map in TRP which we can zoom in and out on our points of interest. We discuss this analysis in Section 6.2.1 through 6.2.6. We use the visual attractiveness of the original plan generated by TRP as benchmark (Figure 43).

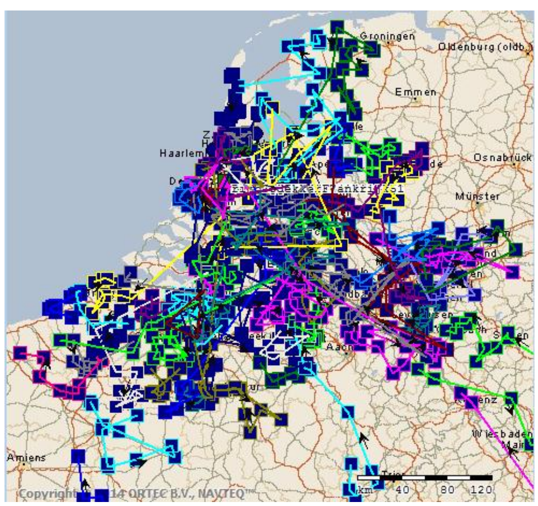


FIGURE 43 - PLAN GENERATED WITH THE ORIGINAL ALGORITHM OF TRP

During the tests, we sometimes stopped the algorithm to look at the visual attractiveness of the plan. This makes it possible to say something about the structure how the plan is created.

Table 21 - Performance indicators approaches

	Original TRP	P_SR	P_LD	S_V1_SR	S_V1_LD	S_V2_SR	S_V2_LD
Total costs	327,657	308,109	301,025	309,496	320,125	303,026	294,833
Total number of vehicles	96	94	92	90	95	89	86
Total number of planned orders	970	970	970	919	967	969	970
Average number of planned orders per trip	10.1	10.3	10.5	10.2	10.2	10.9	11.3
Total number of unplanned orders	0	0	0	51	3	1	0
Total number of planned trolleys	37,115	37,115	37,115	35,507	37,030	37,039	37,115
Total number of trips with violations	0	8	10	0	0	0	0
Total number of kilometers driven	50,549	46,312	46,323	50,302	50,916	46,598	45,793
Average number of kilometers driven per shift	526.6	492.7	503.5	558.9	536.0	523.6	532.5
Total shift duration	1567:27	1464:45	1414:02	1514:10	1538:27	1456:25	1409:24
Average shift duration	16:19	15:34	15:22	16:49	16:11	16:21	16:23
Total driving time	827:14	761:35	759:07	813:50	828:07	766:59	756:12
Total waiting time	97:03	93:08	68:03	70:09	52:00	60:14	48:35
Total number of customers with waiting time	55	43	39	45	38	45	45
Average capacity utilization	84.0%	85.8%	87.7%	85.8%	84.7%	90.5%	93.8%
Maximum capacity utilization	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Minimum capacity utilization	6.5%	30.7%	31.1%	14.1%	5.2%	31.5%	39.1%
Modus capacity utilization	99.3%	96.7%	97.2%	100.0%	99.6%	99.8%	99.1%
Average distance to next order	37.00	29.79	28.73	36.87	36.20	31.17	30.30
Maximum distance to next order	177.00	109.67	83.00	160.00	184.50	67.36	67.36
Minimum distance to next order	8.67	6.17	6.10	6.46	6.46	10.00	6.38
Average total distance between orders	336.71	293.21	298.13	344.19	331.47	340.06	339.73
Maximum total distance between orders	1,038.00	1,038.00	1,079.00	1,155.00	1,038.00	1,038.00	1,038.00
Minimum total distance between orders	17.00	59.00	61.00	43.00	8.00	41.00	53.00
More trips to city than required	62.00	33.00	37.00	44.00	55.00	53.00	55.00
Average radius	1.01	0.81	0.77	1.08	1.06	1.01	1.06
Maximum radius	6.17	6.17	6.17	6.17	6.17	6.17	6.17
Minimum radius	0.00[5]	0.05	0.05	0.00[5]	0.08	0.00[5]	0.08
Total sum of radii	96.41	75.21	70.23	101.37	90.90	88.79	90.90

[5] When the minimum radius is 0, all orders delivered in that trips are in the same city.

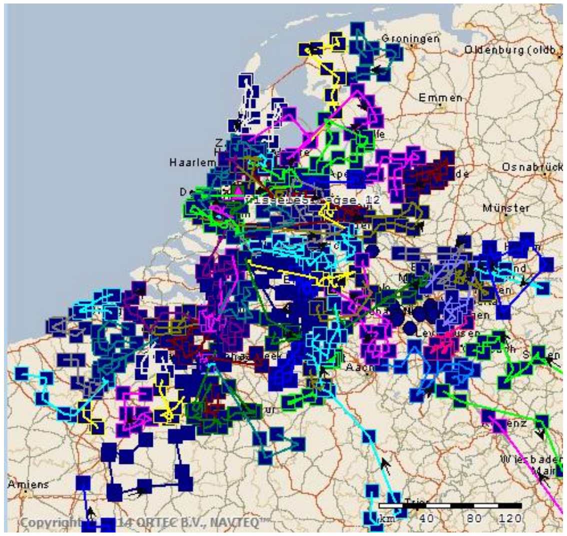


FIGURE 44 - PARALLEL, SMALLEST RADIUS

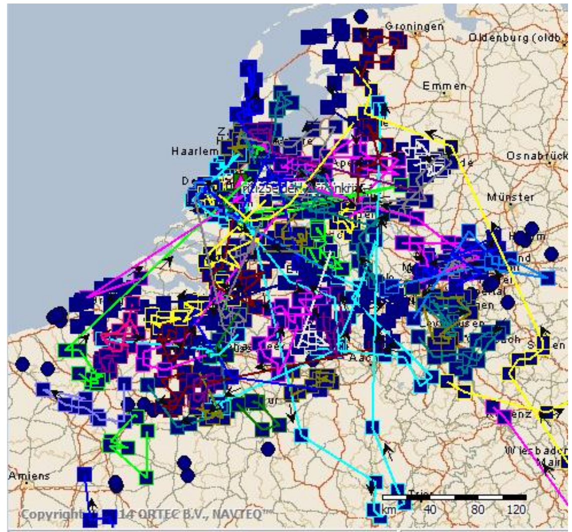


FIGURE 45 - SEQUENTIAL, VEHICLE FIRST, SMALLEST RADIUS

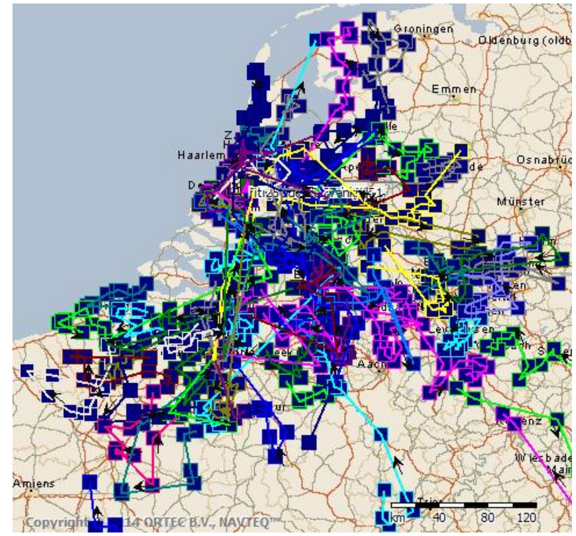


FIGURE 46 - SEQUENTIAL, VEHICLE SECOND, SMALLEST RADIUS

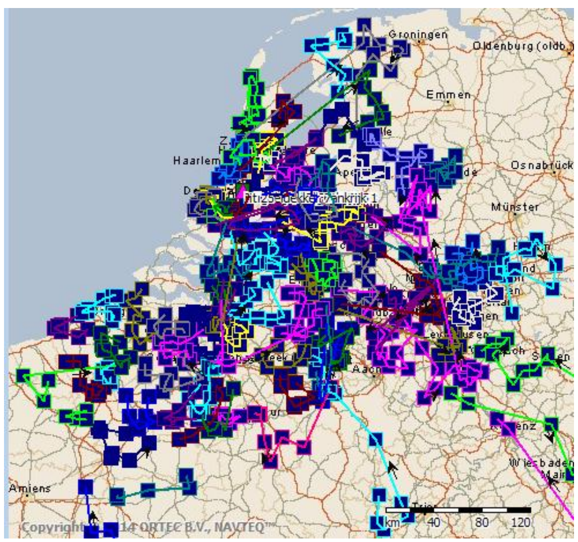


FIGURE 47 - PARALLEL APPROACH, LARGEST DIFFERENCE

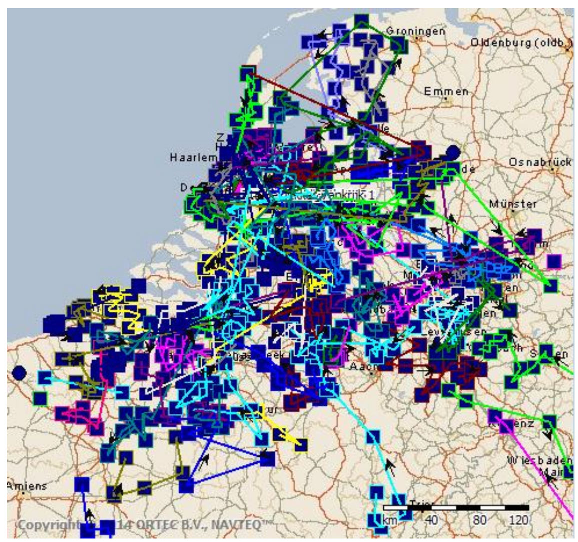


FIGURE 48 - SEQUENTIAL APPROACH, VEHICLE FIRST, LARGEST DIFFERENCE

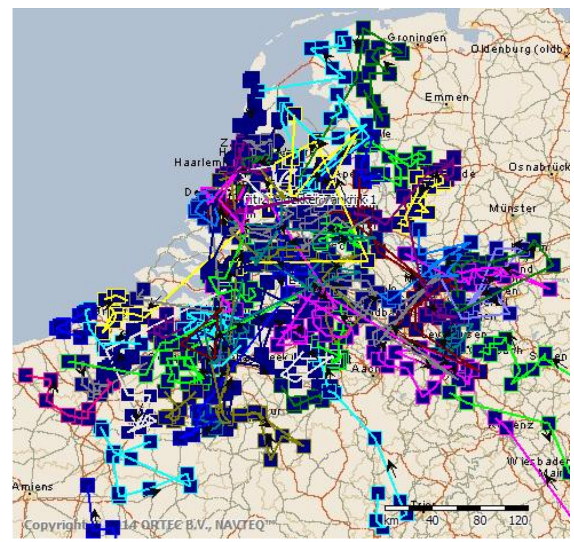


FIGURE 49 - SEQUENTIAL APPROACH, VEHICLE SECOND, LARGEST DIFFERENCE

To get the results of the tests, we generate the initial solution with the new approach and run the improvement steps. In this section, we successively discuss the results of the parallel approaches (Sections 6.2.1 and 6.2.2) and the sequential approaches (Section 6.2.3 through 6.2.6).

We perform the tests with a 4200 trolley dataset of Zeeman. This means we need to plan 970 orders. One truck can transport a maximum of 46 trolleys. For each approach, we compare the plan generated by TRP with the plans from our approaches. We call the plan generated by TRP the original plan.

6.2.1. Parallel approach, smallest radius

We run a test in which we use the parallel approach with the smallest radius as incentive. After the improvement steps, we find that eight of the 94 trips have violations. Four of these violations are related to the shift duration. Three of those four violations overrun the duration with less than half an hour. The planners indicate that it is not a problem if some shifts have slightly longer shift duration than defined. Therefore, we do not pay too much attention to the violations of these three trips. The fourth trip violates the trip duration with 41 minutes. During the shift, there is a waiting time of more than an hour. Probably, the planner will manually adjust the plan by planning one order in another trip such that the duration of this trip is reduced. The other four violations are assigned to the position of the orders in the trip. We discussed the cause of this kind of violation in Section 6.1.2.

The planners should make manual adjustments to make the plan feasible. Therefore, the performance indicators we analyze will not be the final indicators of the plan. This makes the comparison not completely fair. The costs of the plan generated by the parallel approach with the smallest radius incentive, before the manual adjustments of the planners are about 6.0% lower than the costs in the original plan. The other two most important indicators are also reduced; the total number of kilometers with 9.4% and the shift duration with 10.1%. All other indicators are also improved.

Furthermore, we evaluate the specific indicators. The average distance to the next order is decreased almost 20%, which indicates that the solution is more clustered. The average radius of the clusters is reduced even with 25%. The number of cities that is visited by more trucks than necessary is almost reduced by 50%. We find that the capacity utilization of the vehicles is improved to an average of 85.8%. When we investigate the capacity utilization in more detail, we find that 59 vehicles have a capacity utilization of 90% of higher, which is good.

Figure 44 shows the plan of Zeeman for the Netherlands, Belgium, Luxembourg, a part of France, and a part of Germany for this parallel approach. In comparison with the visual attractiveness of the plan generated with the original algorithm of TRP, we find a more clustered solution. There are still a couple of trips that cover a large distance, but we can distinguish the different trips. Especially, in the middle of the Netherlands a large improvement of the visual attractiveness of the plan is apparent.

6.2.2. Parallel approach, largest difference

In the next test, we use the parallel approach with the largest difference selection criterion to generate the initial solution. There are violations of some restrictions in the plan after running the improvement steps. Of the 93 trips, 10 trips give a violation. Five of these violations are overrunning the shift duration with less than a half hour. For the same reasons mentioned in 6.2.1, it is acceptable to ignore these violations. Four of the other five trips have violations due to the sequence of the orders in the trips. In the fifth trip, the shift duration is exceeded due to a waiting action of five hours. The same reservations about comparing this plan to the original plan are warranted here. All indicators are improved in comparison to the indicators of the original plan. The planners will make additional costs to remove the violations. However, violations, such as overwork of the driver, are expensive in TRP. When the planners manually reduce the number of violations, this will have a positive effect on these costs, although it cannot be predicted whether the total costs are increased or reduced.

The average distance to the next order shows a decrease, which indicates a more clustered solution. The average radius of the clusters is reduced by 31.5%. This brings the result very close to the planning manually adjusted by the planners of Zeeman. When we judge the visual attractiveness of the plan (Figure 45), this is confirmed; we can clearly distinguish the different trips in the plan. Most crosses through different areas, which we found in the parallel approach with the smallest radius method, are smaller in this approach. Also the trips in Germany seem more clustered. The number of cities visited by more trucks than necessary is decreased by more than 40% compared to the original plan. The capacity utilization is improved either with to 87.7%.

6.2.3. Sequential approach, vehicle first, smallest radius

We start the validation of the sequential approach with the approach in which we select the vehicle in the first step of the algorithm and then select the seed order with the smallest radius method.

Most striking is that this approach is not able to plan all orders into trips. Even after the improvement steps, 51 orders remain unplanned. This is caused by a lack of vehicles with the required shift for the orders that are unplanned. In most cases, the unplanned orders do not have other orders in the direct neighborhood of their location, which increases the difficulty of connecting them to an existing trip. They are located in France, Germany, or the north of the Netherlands. Since not all orders are planned, comparing the plan on the performance indicators is not very useful; it does not give a fair picture of the quality of the solution. We briefly discuss the specific performance indicators. We find that these are slightly better than the specific performance indicators of the original plan. Especially the performance indicators of the shift work vehicles are quite good. The orders delivered with shift work vehicles are mostly located in the areas with higher density of the orders, which makes it easier to make clusters with a smaller radius.

Visually, the solution is more attractive than the original TRP (Figure 45); we can easier distinguish the different trips. However, there are still crossings in the map. The number of crossings is not very large, but these trips do cross a large region. These are mainly trips that are created later in the process. The algorithm cannot make a full truck load with solely orders in the neighborhood of the location of the seed order. Therefore, orders from further away are assigned to the cluster. We find that most of the trips that cover a larger distance have a seed order outside the Netherlands.

6.2.4. Sequential approach, vehicle first, largest difference

In this test, we run the sequential approach in which we first plan the vehicle and use the largest difference as incentive. We find that the approach not succeeds in planning all orders; three orders remain unplanned. All three unplanned orders are in The Hague and should all be delivered in a Monday night shift. However, there is no vehicle with this shift available when the approach tries to plan these orders.

Since only three orders are unplanned, and the planners probably succeed in making manually adjustments to the planning, such that these orders can be delivered, we evaluate the performance indicators of this approach. We find that these are slightly better than those of the original plan with respect to the costs and the shift duration. Conversely, the total number of driven kilometers is higher. When we contrast the indicators with respect to clustering, we find that the average distance between orders is reduced with 27.7%. We reduced the number of trucks that visit a city more often than necessary with 13%.

Figure 48 depicts the plan we generated with this approach. Visually the plan looks more clustered than the original plan. However, there are also quite some trips that ruin the view by crossing all over the map. These scattered trips are mainly generated in the last couple of runs of the algorithm. There are some orders that become isolated since the orders in the neighborhood of the location of these unplanned orders are planned earlier in the process. The assignment method of TRP plans these orders in one trip, to get close to the capacity of the vehicle in that trip, which causes trips that cover a large distance. The extent to distinguish trips is improved in comparison to the original plan. However, improvements are still possible.

6.2.5. Sequential approach, vehicle second, smallest radius

In this test, we validate the sequential approach in which we select the seed order in the first step and select the vehicle in the second step. We use the smallest radius as incentive.

The first thing we notice is that one order is not planned. This is an order in Belgium with a very specific time window. The trips in the neighborhood of the location of this unplanned order have another shift. Therefore, this order becomes isolated. With some manual adjustments of the planner, it is possible to plan the order in a trip. The performance indicators show an improvement of the plan compared to the original plan. The costs are decreased with 7.5%. The number of kilometers driven is reduced with 7.8% and the shift duration with 1.0%.

From the performance indicators for clustering, we conclude that the average distance to the next customer is reduced as well. The average radius of the clusters remains equal, because there are some extensive clusters that drive through different regions. There are also some very nice clustered trips. The capacity utilization is improved with 6.5%. This is also a result of the reduced number of vehicles needed to plan the orders.

Although there are still some trips that crisscross over the map in the plan, it is possible to distinguish the different trips. This is an improvement to the visual attractiveness of the original plan. The overall visual attractiveness of this plan is not bad.

6.2.6. Sequential approach, vehicle second, largest difference

In the last test, we use the sequential approach in which we successively select the seed order and the vehicle. As incentive, we use the largest difference. With this approach, we are able to plan all orders in feasible trips.

Based on the general performance indicators, this approach gives the best results. We improve all indicators, most of them around 10% compared to the indicators of the original plan. Amongst the improvements are the most important indicators: costs (-10.0%), number of kilometers driven (-9.4%), and number of driving hours (-8.6%).

When we evaluate the specific indicators, we find that the average distance to the next order is decreased with 11.3%. This is an indication that the solution is more clustered. However, the average radius of the trips is slightly higher than in the original plan. When we evaluate the more detailed information, we find that the radius of the trips driven with the shift work vehicles have a better radius than those vehicles in the original plan. The average radius is increased by the short multiple day vehicles. There are still 55 cities that are visited by a truck more times than necessary. Although, this number is lower than in the original plan, it is relatively high in comparison to most other approaches. By evaluating the trips, we find two causes. For the first cause, we evaluate the visual attractiveness of cities that should be visited by two vehicles or more. In these cities, we find one clustered route. However, the remaining orders are more scattered in the border of the city and therefore, in most cases, assigned to different trips. The Hague is a city where we find this behavior (Figure 50). We also find another cause, in which we have a larger city with multiple orders that is visited by one vehicle. However, there is another city, with also more than one order, in the neighborhood. Not all orders in this second city fit into the vehicle. Therefore, that city is visited by two vehicles. Figure 51 gives such as situation. The orders in Arnhem are clustered in one route (an improvement compared to the original plan). However, the vehicle is not full yet. Therefore, three orders in Apeldoorn are added to the trip. Consequently, the other three orders in Apeldoorn need to be delivered with a separate vehicle.

The overall visual attractiveness of the plan is better than the original plan (Figure 49). Especially in Belgium and the Randstad in the Netherlands we gain improvements. However, in the south of the Netherlands it is harder to distinguish the different trips.

6.3. Evaluation of the approaches

In Section 6.2, we validated all approaches. We find that the sequential approach with vehicle selection in the second step and the largest difference as incentive is, next to TRP's current algorithm, the only approach that is able to find a feasible solution. However, this does not makes the other approaches useless. Both parallel approaches and the sequential approach with vehicle selection in the second step and the smallest radius incentive give results worthy of further consideration, albeit with the need for small adaptations.

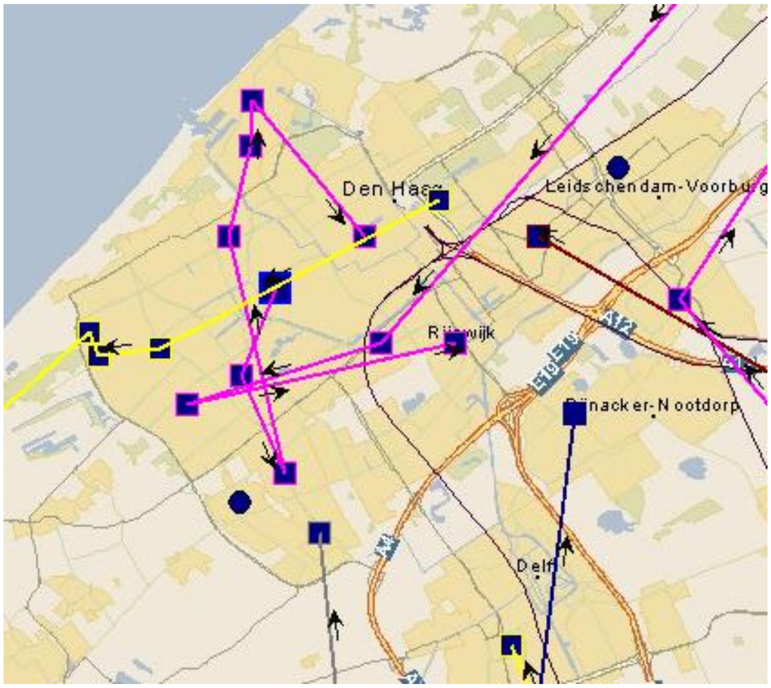


FIGURE 50 – THE HAGUE

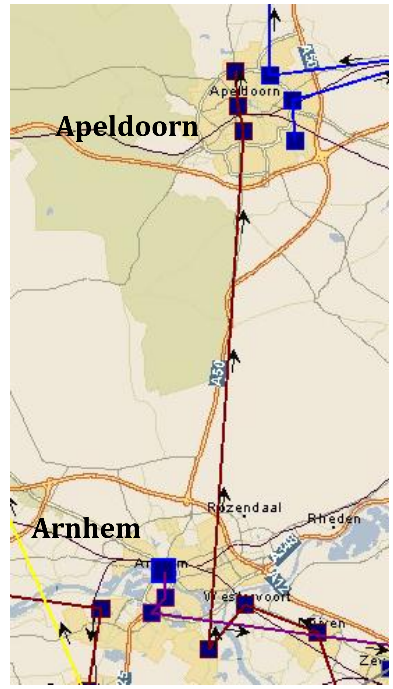


FIGURE 51 - ARNHEM AND APELDOORN

In this section, we successively contrast the parallel and the sequential approaches (Section 6.3.1) and contrast the two incentives: smallest radius and largest difference (Section 6.3.2).

6.3.1. CONTRAST THE APPROACHES: PARALLEL AND SEQUENTIAL

The only feasible solution is created with a sequential approach. However, it would be shortsighted to conclude that this approach is the most promising approach. When we compare the parallel and sequential approach, we find two completely different results. The parallel approach generates a plan in which not all trips are feasible, but which gives good results with respect to clustering and visual attractiveness. The sequential approach generates a plan which scores worse on clustering. However, the reduction in costs and other performance indicators is smaller than in the parallel approach. Three sequential approaches were not able to plan all orders, due to unavailability of the required vehicles.

It is remarkable that the costs of the plans generated with the parallel approach are higher, although the trips are more clustered. The four main input parameters for the costs are the number of kilometers driven, the shift duration, the overwork time, and the fixed costs for the vehicles. When we compare the best plan, based on costs, of the parallel approach (largest difference) and the sequential approach (vehicle second, largest difference), we find that the number of kilometers driven and the shift duration do not differ that much. However, the parallel approach uses six more vehicles, which increases the costs.

In Section 6.2, we validated two variants of the sequential approach. In the first variant, we kept the sequence of the steps equal to the current situation. In the second variant, we first choose a seed order and subsequently choose a vehicle. We find that the latter gives us more freedom to choose an appropriate seed order. In the first variant, we choose a vehicle and choose the best option according to that vehicle. This is not necessarily the best overall option; the chosen seed order may give better results in another shift.

6.3.2. Contrast the incentives: smallest radius and largest difference

In Section 6.2, we tested both incentives with both the parallel and the sequential approach. With both approaches, the largest difference incentive gives the best results. During the seed order selection step, we already consider the vehicle use. Therefore, we avoid the situation that we do not have the required available to deliver the order. Without this check, the shifts may be used for orders that could also be delivered in other shifts.

When we evaluate the parallel approaches, we find that we generate slightly more trips with violations when we use the largest difference incentive. Both approaches show negligible violations of the shift duration. However, the parallel approach with the largest radius incentive gives also some violations on time windows. There is no good explanation for this behavior; we expect that these violations would occur with both incentives. Tests with other datasets must show whether this was a coincidence that there were only violations on shift duration with the smallest radius incentive.

6.4. Conclusion

In this chapter, we answered the question "*Are the approaches an improvement for the planning of Zeeman?*". We answered this question by putting each approach in practice by implementing it in TRP and generating a new plan for a selected dataset of Zeeman.

We first validated the clusters. In Chapter 5, we concluded that the duration of the shift, especially the driving time, is the most uncertain variable in checking the feasibility of the clusters. In Section 6.1.1, we found that the method in which we use 57% of the orders farthest of the seed order gives the driving time closest to the driving time when the sequence of the delivery of the orders is known. The methods in which we use a variable number of orders per cluster gave an insignificant difference with the actual driving time. The method where we used a fixed number showed significant differences. Furthermore, we evaluated the accuracy of the clusters. We concluded that the feasibility check gives not completely a true picture, since we cannot consider the position of an order in the trip. This leads to violations of the time window. We find this back in the results of both parallel approaches. In the sequential approach, we concluded that the assignment method makes other assessments and therefore creates partly different clusters. As the results of the test show, the difference in assessments is not by definition a negative point. We concluded that we generate clusters that can be used in the approaches defined in Chapter 5 and thus the cluster considered valid.

Next, we evaluated the performance indicators and the visual attractiveness of the approaches developed in Chapter 5. Table 22 summaries the most important findings of the validation of the approaches.

Table 22 - Most important findings of validation and comparison

Approach	General performance indicators	Specific performance indicators	Visual attractiveness
Parallel approach, smallest radius	• Four trips with violations. We can ignore three of these violations. • 6.0% lower costs. • Improvements around 10% on other indicators. • Uses two vehicles less.	• Reduction of almost 50% on cities visited more than necessary. • Reduction of the radius (-21%) and average distance to next customer (-20%). • Little worse capacity utilization (+2%).	• Only a few crisscrossing trips. • Possible to distinguish different trips. • Overall picture visual attractiveness: good.
Parallel approach, largest difference	• Ten trips with violations, five violations on time windows due to the position of the order in the trip. • All indicators are improved around 8%. However, manually adjustments of the planners are needed, which will change the indicators. • Uses four vehicles less.	• Reduction of more than 40% on cities visited more than necessary. • Small average radius (reduction of the radius (-24%) and average distance to next customer (-22%). • Little worse capacity utilization (+ 4%).	• Barely crisscrossing trips. • Possible to clearly distinguish different trips. • Overall picture visual attractiveness: very good.
Sequential approach, vehicle first, smallest radius	• Not able to plan 51 orders, due to unavailability of required vehicles. • Indicators are improvement (-3%), although these will get higher after we planned the unplanned orders. • Uses six vehicles less.	• The number of cities visited more than necessary remains equal. • Larger average radius (rise of the radius (+6%) and average distance to next customer (+2%) • Little improved capacity utilization (-2%).	• Crisscrossing trips is improved, but some trips that cover a large distance. • Possible to distinguish different trips • Overall picture visual attractiveness: quite OK.
Sequential approach, vehicle first, largest difference	• Three orders remain unplanned. • Costs (-2%) and shift duration (-	• Reduction of almost 25% on cities visited more than necessary.	• Quite some crisscrossing trips. • No very good score on distinguishing

	2%) slightly better than original plan. • Number of kilometers driven slightly higher (+0.7%). • Uses one vehicle less.	• Small average radius (reduction of the radius (-24%) and average distance to next customer (-2%). • Little improved capacity utilization (-0.8%).	different trips. • Overall picture visual attractiveness: not very good
Sequential approach, vehicle second, smallest radius	• One unplanned order. • Improvements around 7.5% on all indicators. • Uses seven vehicles less.	• Reduction of almost 21% on cities visited more than necessary. • Small improvement on average radius (-0.6%) and improvement of average distance to next customer (-16%). • Improved capacity utilization with almost 8%.	• Some crisscrossing trips, but not very much. • Possible to distinguish different trips. • Overall picture visual attractiveness: good.
Sequential approach, vehicle second, largest difference	• Planned all orders and no trips with violations. • All indicators are improved with about 10%. • Uses ten vehicles less.	• Reduction of 25% on cities visited more than necessary. • Improvement on average radius (-4%) and average distance to next customer (-18%). • Improved capacity utilization with almost 12%.	• Quite some crisscrossing trips, but not very much. • Possible to distinguish different trips. • Overall picture visual attractiveness: better in some regions, worse in others.

With respect to the comparison, we concluded that all suggested approaches are an improvement to the approach currently used in TRP. However, the results of some approaches are more useful than other results, because the results are less promising or the number of (manual) adjustments that the planners need to make is too large. This leads up to conclude that only the sequential approach, with vehicle second generates a complete and feasible solution. Although, both parallel approaches gave some violations, these are still promising approaches with good results on clustering. The parallel variant with the largest difference incentive had slightly better scores on our criteria. From the results, we concluded that the largest difference method is more promising, since the results were better. In the sequential approach, it is important to plan the vehicle in the second step.

7. Conclusion and recommendations

Section 7.1 contains the conclusions that we draw based on this research. In Section 7.2, we have a small discussion about some choices made during this research. We give our recommendations for ORTEC in Section 7.3. We conclude the chapter with suggestions for further research in Section 7.4.

7.1. Conclusion

The goal of this research was to "*Find the cause why the plan generated with TRP is visually less attractive than the plan after the manual adjustments of the planners and develop an improvement of the current planning algorithm used by TRP with a focus on improving the initial solution*". We reached this goal by answering five different research questions. In this section, we briefly discuss the answers to these questions.

We defined indicators that examine the quality of the plan and indicators that specifically judge the extent of clustering in a plan. The four indicators of the latter are:

- the number of cities that are visited by more vehicles than required,
- the average driven distance between the first and the last order in a trip,
- the average radius of the clusters, and
- the average capacity utilization of the vehicles.

We found that on all indicators, the manually adjusted plan of Zeeman scores better than the plan generated by TRP's original algorithm. We concluded that it was not possible to identify one single cause. The most plausible explanation is that the planners explore the neighborhood of the location of the order before inserting the order into a trip, where TRP does not consider this. An important characteristic of the planning that makes it difficult to generate a clustered and feasible plan are the time windows of the orders and the required vehicle types.

When we analyzed the sequential insertion algorithm in the provided cases, we concluded that it is most promising to improve the second step, select the seed order. We developed two approaches: a parallel approach and a sequential approach. In each approach, we use an incentive to generate clusters. The most promising solution found in literature is the circle covering method of Savelsbergh (1990). We used this method as basis for our incentives in the approaches we developed. With this incentive, we explored the neighborhood of the order, before we choose that order as seed order (sequential approach) or choose the cluster (parallel approach). We defined two incentives: the smallest radius and the largest difference between the radii of different shifts.

In the parallel approach, we simultaneously merge the two clusters with the highest incentive. We tested the approach with both the smallest radius and the largest difference incentive. The parallel approach shows multiple strong points on which the plan is improved. With both incentives, the visual attractiveness scores high; the solution looks more clustered. This is confirmed by the performance indicators. With both methods, we get a plan with some violations on mainly trip duration. However, with some manually adjustments, the planners can make the plan feasible.

The second approach is an adjustment to the sequential insertion algorithm. We developed two variants. In the first variant, we only change the seed selection step. We use the smallest radius of a cluster or the largest difference between the radii of different shifts as selection criterion. The variant with the smallest radius does not give a feasible solution; there are too many unplanned orders because the required vehicle was no longer available. With the largest difference incentive, we overcome this problem. Although, there are still some unplanned orders. In the second variant, we again used one of the incentives as selection criterion for the seed, but in the approach, this is the first step of the algorithm and we select the vehicle in the second step. With the largest difference incentive, we generated a completely feasible solution. This approach gave, from all sequential approaches, the best results, both on visual attractiveness and performance indicators.

The parallel approach scores relatively high on clustering, but the costs are relatively high in comparison to the sequential approach. This is mainly caused by the additional number of vehicles the parallel approach needs to plan all orders.

We concluded that we succeed in improving the plan of TRP for the case of Zeeman. There are three valid approaches: both parallel approaches and the sequential insertion algorithm with seed selection as first step and the largest difference incentive. From the parallel approach, the variant with the largest difference gave the best overall result. In the parallel approach, the planners will focus on reducing the number of trips. The sequential approaches scores better on costs and kilometers driven, but it gives a less clustered solution. Most planners will probably prefer using the sequential approach, since improvements based on the visual attractiveness are easier to detect. However, it depends on the preferences of the planners, which plan the will use as starting point of their manual adjustments.

7.2. Discussion

The proposed approach is an improvement for Zeeman. To conclude something about the general applicability of the approach, we need to perform tests with other datasets.

In our approaches, we used zones to force the planning to start with the orders farthest away of the depot. The width of the zones was determined with an educated guess and some small tests. However, no validation was performed whether this is the optimal width of the zones; smaller or larger zones may give other results and thus influences the solution.

In this research, we focused on improving the initial solution in the software of ORTEC. We restricted ourselves with the limitations of the software by trying to find a solution that keeps the logic of the sequential insertion algorithm. However, we limited ourselves with this view. An alternative approach could have been to start with a broader view and subsequently find a way to implement the findings in the software.

7.3. Recommendations

We advise ORTEC to implement the approaches at Zeeman. The planners of Zeeman can extensively test the results on different datasets. In the meanwhile, the approaches can be validated on datasets of other customers, such that ORTEC can implement the approaches at those customers in the future.

We advise to perform additional tests in which they combine the qualities of both approaches. This new approach will be a three-phase approach. One of the strong points of the sequential approach was the small number of trips. Therefore, we first determine a number of seed orders with the sequential approach. Subsequently, we assign orders to the trips of these seed orders with the parallel approach. In this way, we expect to combine the quality of both approaches. We discuss this further in Section 7.4.

As explained in Chapter 3, TRP uses a set of improvement steps to improve the initial solution. The steps we described in Chapter 3 were only a subset of the total available improvement steps. Next to the steps that are used, also the sequence in which the steps are executed determines the quality of the final solution. The steps and the sequence we used in this research is optimized for the case of Zeeman where the original algorithm of TRP was used. Reviewing these improvements steps may improve the planning a little bit further. We recommend ORTEC to review the sequence of the improvements.

7.4. Further research

In Chapter 6, we validated our approaches. In this section, we discuss some interesting topics left for further research. We focus on the two most promising approaches: the parallel approach with the largest difference incentive (in this section shorted to the parallel approach) and the sequential approach with vehicle second and the largest difference incentive (in this section shorted to the sequential approach. If we refer to other approaches, we specifically name these with their full names.

7.4.1. Feasibility check

In our research, we defined certain restrictions for a cluster. However, we already pointed out that the unknown position of an order in a trip is a risk for the feasibility. In the parallel approach, we found that the plan becomes unfeasible because of this research. The clusters we use to determine the seed order are not feasible in the sequential approach either, but since we only use the seed order, this effect is lower. However, the more accurate the clusters are the better the seed order choice and the final solution.

There is a research field that focuses on determining the feasibility of a trip, if the trip contains orders with time windows. It is possible to use a heuristic to determine the chance that two orders must be delivered in the same time slot. If two orders must be delivered in the same time slot and this time slot is not large enough to deliver both orders in, the heuristic indicates that this trip becomes unfeasible. There are also variants that perform a simplified routing check. The accuracy of the clusters with the feasibility check defined in this research is acceptable, but if there is need to improve the accuracy, it should be investigated how we can consider such an approach in our solution, without increasing the computational time too much.

7.4.2. Improvement of the parallel approach

The parallel approach gains good results with respect to clustering. However, the approach needs more vehicles than the other approaches to deliver all orders, which results in higher costs. Using additional vehicles also leads up to a lower capacity utilization. As explained in Chapter 5, the clusters became too large to merge and that leads to additional vehicles.

To reduce the number of trips, we may need a solution that deviates more from the current algorithm used in TRP. The parallel approach had the strong point that is gives a clustered solution by the simultaneous approach of adding orders. However, the sequential insertion algorithm has the strong point to use less vehicles. The strong aspects of those two approaches can be combined in a three phase approach. In the first step, we generate a solution with the sequential approach. We only keep the seed orders and delete the remaining of the plan. These seed orders are all planned in a separate vehicle and are the starting point of the parallel approach. In Chapter 5, we mentioned the risk that with such a method it is determined on forehand which orders are not together in one trip, since they are both seed orders. We partly undermine this by first generating a sequential approach.

7.4.3. Improvement of the sequential approach

To improve the sequential approach even further, we need to focus on gaining a more clustered solution. In the sequential approach, we defined zones such that the algorithm is forced to start with the set of orders farthest away of the depot. These widths of these zones are established with an educated guess. It can be investigated what the optimal width of these zones is.

In Section 7.4.2, we mentioned an approach which deviates more from the current algorithm in TRP for the parallel approach. Also for the sequential approach, there is an idea of improving the plan with a three phase approach. In that approach, we first generate clusters of orders of which is expected that they are delivered in one truck. For example, all orders in one city are in a cluster. The total load of a cluster may be smaller than the capacity of the truck. When we generated those clusters, we run a sequential approach. This approach works similar to the described sequential approach. However, instead of always adding one order, it is also possible to add the predefined clusters. It may be worth investigating whether this leads to a more clustered solution with good performance indicators.

In Chapter 4, we established that the assignment method that TRP currently uses does not worsen the solution. We conclude that with the right seed choice, the orders of one city are planned in one trip. However, this assignment method really depends on the seed order choice and the vehicle choice. In Chapter 6, we established that the main reason for the difference between the cluster generated with our approach (of which we choose the seed) and the trip in TRP with that seed order is the difference in the assignment method. We did not consider changing the assignment method in our sequential approach, because that would not solve the cause of the problem. This is a statement that is still valid. However, another assignment method may improve the solution even more. Therefore, it may be worthwhile to investigate the effects. In Chapter 3, we already introduced some alternative assignment methods.

8. Bibliography

Baker, E. K., & Schaffer, J. R. (1986). Solution Improvement Heuristics for the Vehicle Routing and Scheduling Problem with Time Windows Constraints. *American Journal of Mathematical and Management Sciences*, pp. 261-300.

Bramel, J., & Simchi-Levi, D. (1995, August). A Location Based Heuristic For General Routing Problems. *Operations Research*, pp. 649-660.

Bräysy, O. (2002). Fast local searches for the vehicle routing problem with time windows. *Informaton Systems Opertional Research*, pp. 319-330.

Bräysy, O., & Gendreau, M. (2005). Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms. *Transportation Science, 39*, pp. 104-118.

Bräysy, O., & Gendreau, M. (2005, February). Vehicle Routing Problem With Time Windows, Part II: Metaheuristics. *Transporation science*, pp. 119-139.

CBS. (2012, July 17). *Centraal Bureau van de statistiek*. Retrieved December 3, 2012, from Wegvervoer; kwartaalreeksen goederenvervoer over de weg: http://statline.cbs.nl/StatWeb/publication/?VW=T&DM=SLNL&PA=37234&D1=92-102&D2=0&D3=0,5,10,15,20,25,30,35,%28l-15%29-l&HD=100728-1449&HDR=G1,T&STB=G2

Clarke, G., & Wright, J. (1962, August 30). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, pp. 568-581.

Desrosiers, J., Soumis, F., Desrochers, M., & Sauvé, M. (1986). Methods for routing with time windows. *European Journal of Operational Research*, pp. 236-245.

Dullaert, W., Janssens, G., Sorensen, K., & Vernimmen, B. (2002). New Heuristics For The Fleet Size And Mix Vehicle Routing Problem With Time Windows. *Journal Of The Operations Research Society*, pp. 1232-1238.

European Parliament. (2006, March 15). Verorderning (EG) nr. 561/2006 van het Europees parlement en de raad.

Feo, T. A., & Resende, M. G. (1995). Greedy Randomized Adaptive Search Procedures. *Journal of Global Optimization*, pp. 109-134.

Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for the vehicle routing problem. *Networks*, pp. 109-124.

Gillet, B. E., & Miller, L. R. (1974, April 11). A Heuristic Alogrithm for the Vehicle-Dispatch Problem. *Operations Research*, pp. 340-349.

Ioannou, G., Kritikos, M., & Prastacos, G. (2001). A greedy look-ahead heuristic for the vehicle routing problem with time windows. *Journal of Opertioanl Research Society*, pp. 523-537.

Joubert, J., & Claasen, S. (2006). A Sequential Insertion Heuristic For The Initial Solution To A Constrained Vehicle Routing Problem. *Orion*, pp. 105-117.

Jozefowiez, N., Semet, F., & Talbi, E.-G. (n.d.). Multi-objective vehicle routing problems. *European journal of operational research*, pp. 293-309.

Koskosidis, Y. A., & Powell, W. B. (1992). Clustering algorithms for consolidation of customer orders into vehicle shipments. *Transportation Research*, pp. 365-379.

Langevin, A., & Soumis, F. (1989). Design of multiple-vehicle delivery tours satisfying time constraints. *Transportation Research*, pp. 123-138.

Laporte, G., Gendreau, M., Potvin, J.-Y., & Semet, F. (2000). Classical and modern heuristics for the vehicle routing prorblem. *International transactions in operational research*, pp. 285-300.

Liu, F.-H. F., & Shen, S.-Y. (1999, November). A route-neighborhood-based metaheuristic for vehicle routing problem with time windows. *European Journal of Operational Research*, pp. 485-504.

Newell, G. F., & Daganzo, C. F. (1985, June 11). Design of multiple-vehicle delivery tours - I A Ring-Radial Network. *Transportation Research*, pp. 345-363.

ORTEC BV. (2011). *Functional Description; optimizers at ORTEC.* Gouda: ORTEC.

ORTEC BV. (2012, January). *About ORTEC*. Retrieved May 29, 2012, from ORTEC: http://www.ortec.com/about/company_profile.aspx

ORTEC BV. (2012). *Oplossing: Transport- en Distributieplanning*. Retrieved December 12, 2012, from ORTEC: http://www.ortec.nl/solutions/vehicle_routing_and_dispatch.aspx

ORTEC BV. (2012). *Products Vehicle Routing and Dispatch*. Retrieved December 12, 2012, from ORTEC: http://www.ortec.com/products/ortec_vehicle_routing_and_dispatch.aspx

Osman, I. H., & Laporte, G. (1996). Metaheuristics: a bibliogaphy. *Annals of Operations Research*, pp. 513-623.

Osman, I. H., & Laporte, G. (1996). Metaheuristics: a bibliography. *Annals of Operations Research*, pp. 513-623.

Poot, A., Kant, G., & Wagelmans, A. (2002). A Savings Based Method For Real-life Vehicle Routing Problems. *Journal of the Operations Research Society*, pp. 57-68.

PostNL. (2012, March 7). *Streekpostcodes*. Retrieved July 9, 2012, from Postnl: http://www.postnl.nl/zakelijk/klantenservice/streekpostcodes/

Potvin, J.-Y., & Rousseau, J.-M. (1993). A Parallel Route Building Algorithm for the Vehicle Routing and Scheduling Problem with Time Windows. *European Journal of Operational Research*, pp. 331-340.

Rijksoverheid. (2011, December 13). *Goederenvervoer over de weg*. Retrieved December 3, 2012, from Beleid goederenvervoer over de weg: http://www.rijksoverheid.nl/onderwerpen/goederenvervoer-over-de-weg/beleid-goederenvervoer-over-de-weg

Russell, R. (1995). Hybrid Heuristics for the Vehicle Routing Problem with Time Windows. *Transportation Science*, pp. 156-166.

Savelsbergh, M. (1985). Local Search in Routing Problem with Time Windowds. *Annals of Operations Research*, pp. 285-305.

Savelsbergh, M. W. (1990). A parallel insertion heuristic for vehicle routing with side constraints. *Statistica Neerlandica*, pp. 139-148.

Solomon, M. M. (1987, April). Algorithms for the vehicle routing and scheduling problems with time windows. *Operations Research*, pp. 254-265.

Tan, C., & Beasley, J. (1984, April). A Heuristic Algorithm for the Period Vehicle Routing Problem. *OMEGA*, pp. 497-504.

Toth, P., & Vigo, D. (2002). *The Vehicle Routing Problem.* Philadelphia: Siam.

Twilt, K. (2012, June 21). Planning process of Zeeman. (A. Bosch, Interviewer)

Zhong, Y., & Cole, M. H. (2005, March). A vehicle routing problem with backhauls and time windows: a guided local search solution. *Transporatation research*, pp. 131-144.

Appendix

A. Improvement steps

The current algorithm of TRP works with multiple improvement steps. We explain these steps in Chapter 3. In this appendix we visually display the actions.

Figure 52 - Optimize within trip

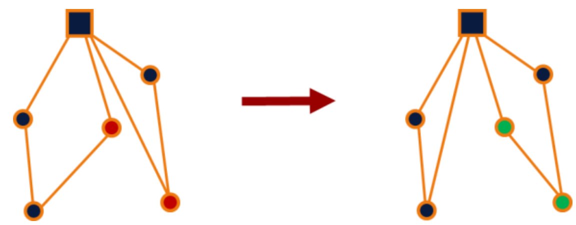


Figure 53 - Move orders between trips

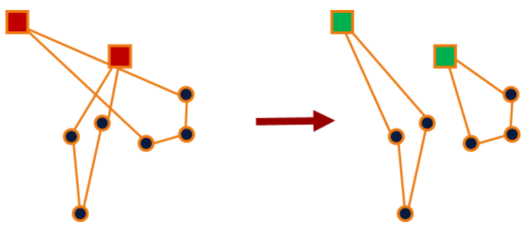


Figure 54 - Exchange vehicles

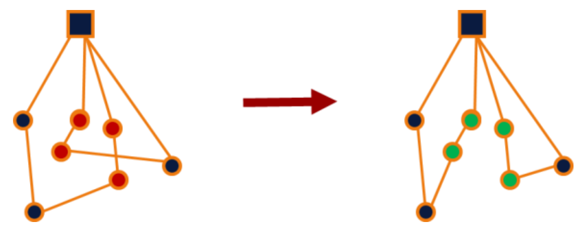


Figure 55 - Exchange orders

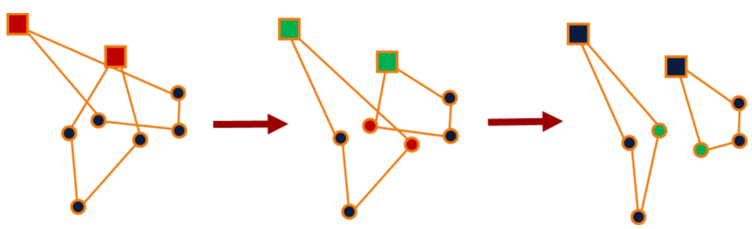


Figure 56 - Optimize between trips

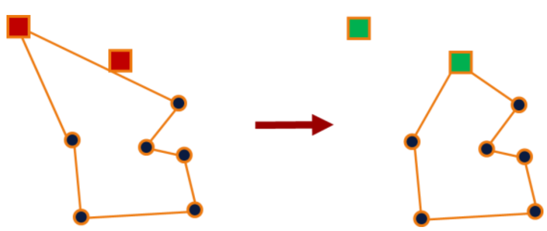


Figure 57 - Choose cheapest vehicle

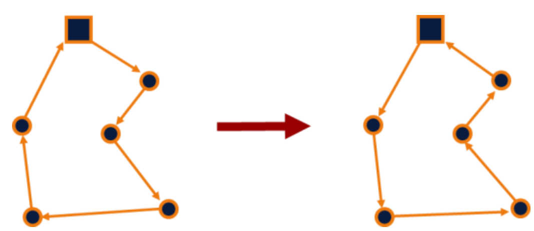


Figure 58 - Flip route

B. Performance indicators plan of Zeeman

	Total	Shift work vehicles	Short multiple day vehicles
Total costs	304,577	145,091	103,449
Total number of vehicles	90	53	28
Total number of planned orders	969	511	324
Average number of planned orders per trip	10.8	9.6	11.6
Total number of unplanned orders	1		
Total number of planned trolleys	37,091	21,467	11,613
Total number of trips with violations	2		
Total number of kilometers driven	48,612	21,250	14,687
Average number of kilometers driven per shift	540.1	400.9	524.5
Total shift duration	1464:47	527:52	536:47
Average shift duration	16:16	9:57	19:10
Total driving time	796:19	360:49	247:04
Total waiting time	55:23	14:06	30:52
Total number of customers with waiting time	42	17	17
Average capacity utilization	89.6%	88.1%	90.2%
Maximum capacity utilization	100.0%	100.0%	100.0%
Minimum capacity utilization	3.3%	3.3%	26.3%
Modus capacity utilization	98.3%	98.3%	94.1%
Average distance to next order	34.88	32.79	33.12
Maximum distance to next order	135.00	135.00	124.50
Minimum distance to next order	8.38	0.00	9.20
Average total distance between orders	351.68	274.63	359.61
Maximum total distance between orders	1,038.00	424.00	638.00
Minimum total distance between orders	43.00	0.00	46.00
More trips to city than required	62.00	-	-
Average radius	1.00	0.70	1.00
Maximum radius	6.17	2.66	2.45
Minimum radius	0.00	0.00	0.05
Total sum of radii	87.81	35.55	28.13

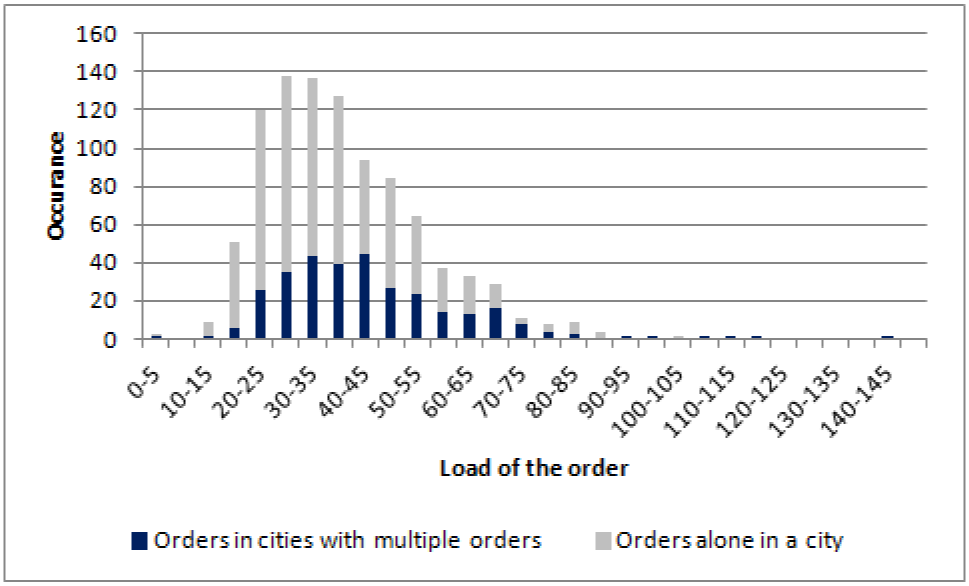
C. Performance indicators plan of TRP

	Total	Shift work vehicles	Short multiple day vehicles
Total costs	304,577	145,091	103,449
Total number of vehicles	90	53	28
Total number of planned orders	969	511	324
Average number of planned orders per trip	10,8	9,6	11,6
Total number of unplanned orders	1		
Total number of planned trolleys	37,091	21,467	11,613
Total number of trips with violations	2		
Total number of kilometers driven	48,612	21,250	14,687
Average number of kilometers driven per shift	540,1	400,9	524,5
Total shift duration	1464:47	527:52	536:47
Average shift duration	16:16	9:57	19:10
Total driving time	796:19	360:49	247:04
Total waiting time	55:23	14:06	30:52
Total number of customers with waiting time	42	17	17
Average capacity utilization	89,6%	88,1%	90,2%
Maximum capacity utilization	100,0%	100,0%	100,0%
Minimum capacity utilization	3,3%	3,3%	26,3%
Modus capacity utilization	98,3%	98,3%	94,1%
Average distance to next order	34,88	32,79	33,12
Maximum distance to next order	135,00	135,00	124,50
Minimum distance to next order	8,38	0,00	9,20
Average total distance between orders	351,68	274,63	359,61
Maximum total distance between orders	1,038,00	424,00	638,00
Minimum total distance between orders	43,00	0,00	46,00
More trips to city than required	62,00	-	-
Average radius	1,00	0,70	1,00
Maximum radius	6,17	2,66	2,45
Minimum radius	0,00	0,00	0,05
Total sum of radii	87,81	35,55	28,13

D. Performance indicators plan generated with savings algorithm

	Total	Shift work vehicles	Short multiple day vehicles
Total costs	321,292	145,091	103,449
Total number of vehicles	95	53	28
Total number of planned orders	966	511	324
Average number of planned orders per trip	10.2	9.6	11.6
Total number of unplanned orders	4		
Total number of planned trolleys	36,940	21,467	11,613
Total number of trips with violations	0		
Total number of kilometers driven	46,981	21,250	14,687
Average number of kilometers driven per shift	494.5	400.9	524.5
Total shift duration	1490:15	527:52	536:47
Average shift duration	15:41	9:57	19:10
Total driving time	770:19	360:49	247:04
Total waiting time	103:22	14:06	30:52
Total number of customers with waiting time	50	17	17
Average capacity utilization	84.5%	88.1%	90.2%
Maximum capacity utilization	100.0%	100.0%	100.0%
Minimum capacity utilization	12.6%	3.3%	26.3%
Modus capacity utilization	94.3%	98.3%	94.1%
Average distance to next order	32.89	32.79	33.12
Maximum distance to next order	130.25	135.00	124.50
Minimum distance to next order	5.57	0.00	9.20
Average total distance between orders	306.25	274.63	359.61
Maximum total distance between orders	1,038.00	424.00	638.00
Minimum total distance between orders	39.00	0.00	46.00
More trips to city than required	30.00	-	-
Average radius	1.06	0.70	1.00
Maximum radius	6.17	2.66	2.45
Minimum radius	0.09	0.00	0.05
Total sum of radii	99.65	35.55	28.13

E. Distribution of the load



Number of trolleys per order	Orders alone in a city	Orders in cities with multiple orders
0-5	2	1
5-10	0	0
10-15	7	2
15-20	45	6
20-25	94	26
25-30	103	35
30-35	94	43
35-40	88	39
40-45	50	44
45-50	57	27
50-55	41	23
55-60	23	14
60-65	20	13
65-70	13	16
70-75	3	8
75-80	4	4
80-85	6	3
85-90	4	0
90-95	0	1
95-100	0	1
> 100	1	4

F. Performance indicators plan with smallest time window seed selection

Note that these results are generated with a preliminary test and therefore cannot be one-to-one be compared with the indicators of the other tests.

	Total	Shift work vehicles	Short multiple day vehicles
Total costs	304,342	145,091	103,449
Total number of vehicles	89	53	28
Total number of planned orders	936	511	324
Average number of planned orders per trip	10.5	9.6	11.6
Total number of unplanned orders	34		
Total number of planned trolleys	35,704	21,467	11,613
Total number of trips with violations	0		
Total number of kilometers driven	46,786	21,250	14,687
Average number of kilometers driven per shift	525.7	400.9	524.5
Total shift duration	1421:29	527:52	536:47
Average shift duration	15:58	9:57	19:10
Total driving time	761:37	360:49	247:04
Total waiting time	56:15	14:06	30:52
Total number of customers with waiting time	40	17	17
Average capacity utilization	87.2%	88.1%	90.2%
Maximum capacity utilization	100.0%	100.0%	100.0%
Minimum capacity utilization	19.1%	3.3%	26.3%
Modus capacity utilization	100.0%	98.3%	94.1%
Average distance to next order	30.04	32.79	33.12
Maximum distance to next order	100.91	135.00	124.50
Minimum distance to next order	6.08	0.00	9.20
Average total distance between orders	302.83	274.63	359.61
Maximum total distance between orders	1,128.00	424.00	638.00
Minimum total distance between orders	53.00	0.00	46.00
More trips to city than required	23.00	-	-
Average radius	1.01	0.70	1.00
Maximum radius	7.24	2.66	2.45
Minimum radius	0.08	0.00	0.05
Total sum of radii	88.99	35.55	28.13

G. Performance indicators plan generated with algorithm for geographical location

	Total	Shift work vehicles	Short multiple day vehicles
Total costs	304,577	145,091	103,449
Total number of vehicles	90	53	28
Total number of planned orders	969	511	324
Average number of planned orders per trip	11	10	12
Total number of unplanned orders	1		
Total number of planned trolleys	37,091	21,467	11,613
Total number of trips with violations	2		
Total number of kilometers driven	48,612	21,250	14,687
Average number of kilometers driven per shift	540	401	525
Total shift duration	61	22	22
Average shift duration	1	0	1
Total driving time	33	15	10
Total waiting time	2	1	1
Total number of customers with waiting time	42	17	17
Average capacity utilization	1	1	1
Maximum capacity utilization	1	1	1
Minimum capacity utilization	0	0	0
Modus capacity utilization	1	1	1
Average distance to next order	35	33	33
Maximum distance to next order	135	135	125
Minimum distance to next order	8	0	9
Average total distance between orders	352	275	360
Maximum total distance between orders	1,038	424	638
Minimum total distance between orders	43	0	46
More trips to city than required	62	0	0
Average radius	1	1	1
Maximum radius	6	3	2
Minimum radius	0	0	0
Total sum of radii	88	36	28

H. Performance indicators developed approaches – shift work vehicles

	Original TRP	P_SR	P_LD	S_V1_SR	S_V1_LD	S_V2_SR	S_V2_LD
Total costs	147,715	140,694	140,477	123,583	147,390	147,261	136,571
Total number of vehicles	56	53	53	46	55	55	51
Total number of planned orders	548	520	526	474	522	542	532
Average number of planned orders per trip	9.8	9.8	9.9	10.3	9.5	9.9	10.4
Total number of unplanned orders	0	0	0	0	0	0	0
Total number of planned trolleys	22,195	20,894	21,122	19,062	22,499	22,193	21,525
Total number of trips with violations	0	0	0	0	0	0	0
Total number of kilometers driven	17,894	17,473	18,116	16,268	19,451	19,399	18,066
Average number of kilometers driven per shift	319.5	329.7	341.8	353.7	353.7	352.7	354.2
Total shift duration	517:16	508:45	503:12	447:33	523:00	538:56	493:00
Average shift duration	9:14	9:35	9:29	9:43	9:30	9:47	9:40
Total driving time	319:29	310:10	315:38	291:16	335:32	338:53	322:38
Total waiting time	39:59	33:27	17:20	21:42	28:38	23:58	16:46
Total number of customers with waiting time	21	18	12	18	20	21	22
Average capacity utilization	86.2%	85.7%	86.6%	90.1%	88.9%	87.7%	91.8%
Maximum capacity utilization	100.0%	100.0%	99.8%	100.0%	100.0%	100.0%	100.0%
Minimum capacity utilization	19.1%	30.9%	32.8%	24.8%	38.9%	31.5%	39.1%
Modus capacity utilization	97.6%	96.7%	99.8%	100.0%	99.6%	99.1%	99.1%
Average distance to next order	30.34	33.68	30.04	35.99	43.99	34.11	32.41
Maximum distance to next order	50.14	84.50	57.20	100.00	184.50	53.11	51.30
Minimum distance to next order	15.00	6.56	8.33	10.56	8.00	18.79	10.57
Average total distance between orders	311.22	305.72	303.17	310.79	356.74	382.52	368.19
Maximum total distance between orders	511.00	580.00	912.00	507.00	606.00	521.00	521.00
Minimum total distance between orders	199.00	59.00	75.00	43.00	8.00	79.00	74.00
More trips to city than required	1.46	2.44	2.13	2.92	2.27	1.90	2.27
Average radius	17.87	23.51	20.21	33.58	28.61	23.91	28.61
Maximum radius	-	-	-	-	-	-	-
Minimum radius	-	-	-	-	-	-	-
Total sum of radii	-	-	-	-	-	-	-

I. Performance indicators developed approaches – short multiple day vehicles

	Original TRP	P_SR	P_LD	S_V1_SR	S_V1_LD	S_V2_SR	S_V2_LD
Total costs	92,826	111,378	103,721	122,065	116,698	99,728	102,225
Total number of vehicles	23	32	30	34	31	25	26
Total number of planned orders	252	316	309	334	311	293	304
Average number of planned orders per trip	11.0	9.9	10.3	9.8	10.0	11.7	11.7
Total number of unplanned orders	0	0	0	0	0	0	0
Total number of planned trolleys	9,538	12,210	11,964	13,104	10,520	10,835	11,579
Total number of trips with violations	0	0	0	0	0	0	0
Total number of kilometers driven	13,720	16,164	14,978	18,654	18,790	14,524	15,052
Average number of kilometers driven per shift	596.5	505.1	499.3	548.6	606.1	581.0	578.9
Total shift duration	454:25	555:52	502:15	610:25	615:19	517:21	516:16
Average shift duration	19:45	17:22	16:44	17:57	19:50	20:41	19:51
Total driving time	219:54	262:59	247:56	297:47	304:09	239:40	245:08
Total waiting time	13:24	49:16	40:18	38:34	12:57	25:51	21:24
Total number of customers with waiting time	14	17	19	19	10	16	15
Average capacity utilization	90.2%	82.9%	86.7%	83.8%	73.8%	94.2%	96.8%
Maximum capacity utilization	100.0%	100.0%	150.0%	100.0%	100.0%	100.0%	100.0%
Minimum capacity utilization	51.3%	30.7%	31.1%	14.1%	5.2%	50.4%	80.7%
Modus capacity utilization	97.8%	96.7%	97.8%	100.0%	99.6%	99.8%	100.0%
Average distance to next order	24.27	23.65	23.62	26.98	28.94	26.40	25.37
Maximum distance to next order	86.40	109.67	79.00	84.50	78.80	56.14	56.60
Minimum distance to next order	6.08	6.17	6.10	0.00	0.00	10.00	6.38
Average total distance between orders	205.98	205.77	209.87	252.04	243.30	251.44	250.41
Maximum total distance between orders	432.00	405.00	399.00	478.00	455.00	419.00	404.00
Minimum total distance between orders	53.00	64.00	61.00	0.00	0.00	41.00	53.00
More trips to city than required	2.18	3.15	1.88	2.40	1.84	2.88	1.84
Average radius	29.49	27.57	24.66	38.64	36.02	40.75	36.02
Maximum radius	-	-	-	-	-	-	-
Minimum radius	-	-	-	-	-	-	-
Total sum of radii	-	-	-	-	-	-	-