Order pairing strategies for the Petrol Station Replenishment Problem

A model to find new approaches for pairing orders in a multi–period vehicle routing problem with time windows.

J.Y. Brandt

Technische Universiteit Delft

Order pairing strategies for the Petrol Station Replenishment Problem

A model to find new approaches for pairing orders in a multi–period vehicle routing problem with time windows.

by

Jelmer Yannic Brandt

Student number:	1540769	
Email:	J.Y.Brandt@student.tudelft.nl	
Date:	Delft, June 30, 2017	
Thesis committee:	Prof. dr. ir. L. Tavasszy,	TU Delft, TPM, supervisor
	Dr. J. H. R. van Duin,	TU Delft, TPM
	Ir. M. B. Duinkerken,	TU Delft, 3mE
	Ir. K. Hauge,	AMCS group

Keywords: Petrol station replenishment; Vehicle restrictions; Vehicle routing and scheduling; Fuel delivery; time windows.

Preface

This thesis is the final product before finishing my master program of Transport, Infrastructure and Logistics at the Delft University of Technology. The topic of this research is the development of a model to test new order pairing strategies for fuel delivery to petrol stations. The model consists of two separate models, of which one is an optimization model to allocate vehicles and orders on certain routes with the aim to minimize travel times. For the development of the optimization model I had to step out of my comfort zone, since I needed to learn programming in Python and working with optimization software. To learn both in just a few months turned out to be quite challenging, but nonetheless it was like a big puzzle to solve, which I enjoyed very much.

I would like to thank AMCS and Kristian Hauge who provided me with this very interesting topic and gave me the possibility to conduct the research at their office. Whenever I needed help or information, Kristian was always there to give me the right answer. Special thanks go to Leon Driesse who brought me in contact with AMCS and helped me finding my way in the organization. The atmosphere in the office made me feel right at home and I really enjoyed working there.

Furthermore I would like to thank my graduation committee. Mark Duinkerken, for guiding me in the model building process, providing me with detailed feedback and showing me the pitfalls of a rookie in the optimization world. Ron van Duin, for always thinking with me, pointing me in the right direction by helping me how to approach and structure my project. Lóri Tavasszy, for supervising the quality and process of this research during official meetings and showing the same enthusiasm for the topic as I did.

Last but not least, I would like to thank my parents and sister for supporting me during my study time, who always respected my choices during my time here in Delft. Of course I also want to thank my friends and roommates who gave color to my student life. Thanks for these unforgettable years.

Jelmer Brandt
Delft, June 30, 2017

Executive summary

Oil and gas are a vital part of the worldwide economy and will most likely keep an important role for the upcoming decades. Delivery of crude oil as well as high inflammable fuels to petrol stations are a very valuable business. Transportation is one of the things that could be improved to cut costs and decrease the impact on the environment.

This research explores the possibilities to improve order allocation for petrol delivery, with the aim to lower the travel distances. With this information, the following research question has been formulated:

"How can the current order pairing strategy be improved and what impact does this have on the total kilometers driven and the carbon footprint?"

The aim of this research is to design a model that can test different order pairing strategies for fuel delivery to petrol stations. By comparing the characteristics of a case study with the problems described in the literature a research gap was found. The problem was categorized as a Petrol Station Replenishment Problem (PSRP). These problems have vehicles with a limited carrying capacity for the goods that have to be delivered, which are tried to be solved to optimality. The specific combination of station restrictions, time windows, vehicle schedules and a heterogeneous limited fleet have not been implemented until this research. Furthermore most PSRP are only looking at a single–day period, whereas the research in this thesis is aiming to look for a longer time–horizon.

This knowledge gap has led to the implementation of a simulation model, which is the result of the interaction between two sub models. The first one is the *order generation model*, which makes orders by forecasting the levels for all tanks at a station, aiming at equal depletion times.

The second model uses the output of the first model. This *order pairing optimization model* is an extension to the petrol station replenishment problem with the objective to minimize travel times. The Multi–Period Split Compartment Vehicle Routing Problem with Time Windows and Vehicle Restrictions (MP-SCVRPTWVR) has been compared and validated using an test instance from Coelho and Laporte (2015).

The model could reach optimality for up to 20 stations. However, having 20 or more stations became more challenging for the model to solve. A proposition was suggested by going from an unsplit model towards a split model, which in fact are two separate optimization models. The first part of the split model assigns routes to vehicles, aiming at minimizing travel times. The second model will use these routes to define the amounts that have to be loaded into the vehicle compartments. The split model showed to be a very promising alternative for the unsplit model, as the route assignment model returned equal outcomes. Furthermore the calculation times were greatly reduced and it was possible to solve larger instances, reaching optimal solutions in two hours for runs with 25 stations.

With the split models integrated in the simulation model, four different strategies have been tested. One of the conclusions that can be drawn is that the user should not focus on making as much paired orders as possible. In all strategies this turned out to increase the travel distances, while the

aim is to minimize them. Furthermore it can be concluded that it might be rewarding to try grouping stations on certain aspects. This turned out to be one of the most promising outcomes while testing strategies in this research. The second option is to look at the way orders are allocated in the trucks. This could be in different sequences, for example starting with half of the order amount for all stations, or start with full order amounts for the station with high consumption. There are a lot of different combinations possible which need future research.

In conclusion, the model presented is applicable for sample sizes of up to 50 to 55 stations if the order allocation is equally divided over the time period. This gives enough options to test different strategies using this simulation model. For one day runs the order pairing optimization model can handle up to 26–27 stations per day if the split model is used. When the unsplit model is used, this falls back to 19–20 stations.

Contents

List of Figures

List of Tables

List of symbols

i	Index of a gas station i, belonging to set $V = \{0, 1, ..., n\}$ with the depot having index 0.
r	Index of route r, belonging to feasible routes set $R = \{0, 1, ..., r\}$ where r is a subset $V_r \subseteq V$.
k	Index of the vehicle k, belonging to set $K_1 \cup K_2 = \{1, 2, ..., k\}$.
c	Index of the compartment c, belonging to set $C_1 \cup C_2 \cup C_2^{'} = \{1, 2, ..., c\}$, where $C_2^{'} \subset C_2$.
p	Index of station fuel tank p, belonging to set $P = \{1, 2, ..., p\}$.
Q^{ck}	Capacity of compartment c of vehicle k.
L_i^{pt}	Stock level of tank p of station i at the end period t.
L_i^{p0}	Initial stock level of tank p of station i.
u_i^{pt}	Consumption rate of tank p of station i at period t.
t_i^k	The earliest time at which delivery of vehicle k at station i can take place.
$t_i^{d,k}$	The latest time at which delivery of vehicle k at station i can take place.
$v_{min,i}^p$	Minimum delivery amount for tank p of station i.
$v_{max,i}^p$	Maximum delivery amount for tank p of station i.
α_r	The earliest departure time for route r.
β_r	The latest departure time for route r.
C_i^p	Maximum capacity of tank p of station i.
$C_{min,i}^p$	Minimum capacity of tank p of station i.
$[t_i^a, t_i^b]$	The delivery time window of station i.
W_{start}^{kt}	Start time of the work schedule W of vehicle k at time period t.
W_{end}^{kt}	End time of the work schedule W of vehicle k at time period t.
H_i^t	Opening hours of station i at time period t.
S	The delivery rate, which indicates the replenishment time when a vehicle arrived at a station.
t_{ij}	Travel time from station i to station j.
M	Arbitrary large value.

d_r^k	The time vehicle k departs from the depot at the start of route r.
a_i^p	Amount a of fuel type p delivered to station i.
v_k	Integer value being zero if one route is assigned to vehicle k and receives +1 for every additional route on the same vehicle. If a vehicle is not used this variable becomes $-\infty$.
q_{ir}^{pck}	The quantity of product p delivered to station i of route r in compartment c of truck k.

$$x_r^k = \begin{cases} 1, & \text{if vehicle } k \text{ is using route } r. \\ 0, & \text{otherwise.} \end{cases}$$

$$R_i^k = \begin{cases} 1, & \text{if vehicle } k \text{ is allowed to visit station } i. \\ 0, & \text{otherwise.} \end{cases}$$

$$z_{ir}^{pck} = \begin{cases} 1, & \text{if station } i \text{ receives product } p \text{ from compartment } c \text{ of vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$$

$$s_{rs}^k = \begin{cases} 1, & \text{if route } s \text{ follows route } r \text{ for vehicle } k. \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{ir}^k = \begin{cases} 1, & \text{if station } i \text{ is visited by vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$$

$$w_r^{pck} = \begin{cases} 1, & \text{if product } p \text{ is loaded in compartment } c \text{ of vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$$

$$l_{r,W}^k = \begin{cases} 1, & \text{If route } r \text{ is in work schedule } W \text{ of vehicle } k. \\ 0, & \text{otherwise.} \end{cases}$$

I

Problem analysis

1

Introduction to the petrol station delivery problem

Oil and gas are a vital part of the worldwide economy and will most likely keep an important role for the upcoming decades. As oil gets more and more valuable due to increasing demand and a decreasing amount of sources, it is inevitable that prices will increase throughout the years. To be able to control the price increases, petroleum companies are looking for ways to decrease costs. One of the options is to improve the supply chain, where the Logistics are an important factor. This definitely is the case for the petroleum industry, where transportation of crude oil as well as high inflammable fuels to petrol stations are a very valuable business. This leads to high logistics costs, making it an important factor to look at when trying to improve the supply chain (Varma, Wadhwa, & Deshmukh, 2008).

Transportation is one of the things that could be improved to cut costs and prices at the petrol stations. While it has obvious economical benefits for society to cut transportation costs, the petroleum companies themselves have benefits from it as well. Since 2014 fuel prices have seen a sharp decline due to overproduction and the aftereffects of the 2008 global financial crisis. This has led private companies such as Shell and BP to heavily cut their costs in exploration and other investments (Rogoff, 2016). Transportation is something that is difficult to change, as a certain amount of orders need to be handled regardless of the oil prices. In fact, the global oil consumption increases every year independent of prices. This will lead to increased size and amount of orders, thus more transportation is needed which gives another reason to try and optimize the petrol deliveries.

Costs are not the only incentive for the petroleum industry to be looking for improvement. The petrol industry is gaining more and more pressure from the public opinion, which demands more responsibility from the companies to protect the environment. As transportation has a large impact on the carbon footprint of a company, one way to decrease the impact on the environment is by improving distribution efficiency and environmental safety (Transvision, 2014).

AMCS group, a company specializing in routing and planning solutions, provides software to businesses that can create real–time planning and scheduling. They aim to keep their status as being one of the leading providers of planning en scheduling software, which makes it important to constantly improve their software. Mostly focusing on the waste and recycling branch, oil companies are also among the customers making use of their services and AMCS has an algorithm running in their software specifically focusing on petrol delivery.

Since the transport is a costly part in the supply chain and the petroleum industry is emphasizing more on improving the transport efficiency, optimization of petrol order allocation is an interesting subject to which AMCS is looking for improvement. More specific, they would like to look for possible improvements in their order pairing algorithm. This part of the model uses the input from the incoming orders to look for orders that can be paired based on similarities in time windows, locations and amounts. They stated that their current order pairing strategy is working and has led to improvements for their customers, but the model has never been further tested with different strategies. Using different strategies could lead to better solutions and lower calculation times, since the order pairing is one of the hardest parts and computing times can get very large when the amount of orders is increasing. Furthermore a significant part is still controlled by human interaction, who make decisions on how many orders are allowed to be paired per truck and if certain combinations of stations on a route is allowed.

This thesis will look into the possibilities for testing new order pairing strategies to get a more efficient planning for the fuel delivery to petrol stations. The report is structured into three separate parts. The first part will give an analysis of the problem of AMCS in the current system. By analyzing the current order strategy and comparing this to the literature it is possible to see where improvements are possible and how to further approach the problem. The second part will be focusing on the model design that is used to test the different strategies. This includes verification and validation of the model. The last part will test several strategies in the model and present the outcomes. Finally conclusions and recommendations are drawn.

1.1. Problem context

Leading from the introduction it becomes clear that the underlying problem is initiated by the petroleum industry to decrease the transportation costs and impact on the environment. AMCS wants to contribute to this by exploring the possibilities to improve their current order allocation strategy for petrol delivery. This research will be an exploration to learn about possible improvements for the current order pairing strategy, with the aim to decrease the amount of kilometers driven. Travelled distances do have a large impact on the transportation costs, but it is also worth to take notice of the so-called lean and green award. This is a green certification given to a company when it manages to emit 20% less CO2 (Connekt, n.d.). While this is not the main objective, it would be nice to compare different strategies on their capability to reduce emissions and if this could lead to a lean and green certificate if adapted in real life.

The current algorithm behind the order pairing of AMCS makes pairs by looking at orders that have the same delivery times. Furthermore the algorithm determines which pairs are "allowed" through the input it gets from the human user. Some of the parameters are partly controlled by parameters and others are controlled by human interaction. The user determines constraints involving proximity between stations, cut down percentages, maximum detour distances, maximum orders allowed in one order pairing, grouping of stations and priority stations. This means a lot of the current planning is actually controlled by the planners. To be able to form good order pairings it means that the user needs to know each scenario by large details. This will become more difficult when a larger and more heterogeneous area is involved. The amount of uncertainties make it very difficult for the user to have a good overview of the total system and this might lead to unnecessary losses of time and money.

While it does not mean that the current strategy is particularly good or bad, AMCS wants to look for possible improvements by trying out new ways to handle orders. It is known that the current strategy can make good schedules for one day periods, but it is not known how that works out over a longer time span. It could well be the case that over a simulation period of a few weeks, another strategy performs better than the current one, while this would not be the case when looking at a single day. The various parameters involved could greatly influence the effectiveness of different order pairing strategies. AMCS wants to explore the possibilities for improvement of the current algorithm used for order pairing. With the aim to lower the travel distances, the following research question has been formulated:

"How can the current order pairing strategy be improved and what impact does this have on the total kilometers driven and the carbon footprint?"

1.2. Research questions and research objective

With the main question defined, the research objective is to design a model that can test different order pairing strategies for fuel delivery to petrol stations. Before getting to the point of the actual model design and trying out new strategies to give an answer to the main question, there are a few other factors that have to be determined. Therefore several sub questions have been defined which should be answered in a step wise approach, which help to analyze the system and derive a suitable model to be used for testing. Eventually the answers on the sub questions will be used to answer the main question. The sub questions are defined as follows:

a. How does the case data provided by AMCS look like and how can it be used in the model?

b. How does the current order pairing algorithm of AMCS look like?

c. What comparable characteristics and models are available in the literature?

c. What KPIs are relevant for measuring the effects of different order pairing strategies and how are they measured?

d. How should the structure and mathematical representation of the model design look like?

e. What strategy improvements could lead to a performance increase compared to the current situation and what are the results?

1.3. Scope of the order pairing problem

Before going further into details, it is necessary to look at the scope of this research. This includes defining clear boundaries on what factors should be accounted for and which can dropped. This research will only focus on the specific case given by AMCS. According to the data, around 60 stations are served from one depot. Thus the boundaries of the model itself are limited by the given data as the depot at one side and the stations at the other side. The number of depots is limited to one, so one depot has to provide fuel to a certain amount of stations.

Since it must be tested if the total mileage and carbon footprint are getting lower with certain strategies, doing so is only possible when you test under the same circumstances. It cannot be the case that some stations are just omitted or vehicles are made up in one strategy while this is not the case for other strategies. Furthermore unexpected factors like traffic jams are not taken into account. The data has provided a travel time matrix, a distance matrix and coordinates using Euclidean space as well as geocoded coordinates. For this research only the two matrices will be used. Using the locations of the stations and depot on a real map and implementing this on a real road network by using geo-coding takes a lot of extra time and does not have much added value in this research, since the matrices provided by AMCS are actually based on their own interpretation of the coordinates.

This research will not specifically take into account usability for the end–user, the clients of AMCS. For the customer of AMCS usability obviously is considered important by them when making a planning. When the order plans are made they must make sense to the end–user. The planner as well as the driver in the vehicle should be able to grasp why certain routes are made. If the planner (or the model) makes a certain order plan it must intuitively feel right to the driver as it can really influence his view on efficiency. Cold hard numbers are logical to the planner maybe, but it can lose its logic in the feeling of the driver. Although this may be important to the customer of AMCS, it is considered out of scope in this research since the focus is put upon improving the order pairing strategies for AMCS itself, creating a tool for them with which they can test certain strategies. Furthermore it is required to keep calculation times as low as possible, which also will be included as one of the factors that can be tested with this model. For example, some strategy might be having less desirable outcomes in terms of travel distances, but when calculation is twice as fast compared to another strategy, it might eventually turn out better cost-wise.

1.4. Research framework

To answer the research question a framework has been made which is shown in figure 1.1. This framework is closely related to the *'engineering design'* methodology as described in Dym, Little, and Orwin (2014). The structure of the report can be divided into three separate parts. *Part one: problem analysis*, will first analyze the problem from AMCS's point of view, which has already been done in the previous section. The next step (chapter 2) will analyze the case data provided by AMCS and see how their current implemented algorithm looks like. This helps to categorize the problem and compare it to relevant literature, finding possible research gaps, which is done in chapter 3.

Based on the research gap, the methodology in *Part two: methodology*, shall further explain the model design, starting with a conceptual design in chapter 4. Chapter 5 will explain more about the mathematical functions used in the order pairing optimization model. Furthermore a route selection procedure is constructed. After that the model will be tested, doing several verification runs and a comparison with another model from the literature for validation. After that the usability and efficiency of the model is tested. Chapter 6 will describe the simulation model, integrating the optimization model together with an order generation model. Furthermore an adaption to the original optimization model is proposed. The simulation model will then be tested and compared with simulation results from AMCS. Lastly a sensitivity analysis is performed on the target amounts.

Part three: Strategies, results and conclusions, the concluding part of this thesis, defines four strategies to be tested in the model. Based on the results in chapter 7, chapter 8 will give conclusions and recommendations, where the main question is answered and implications in both models will be discussed.

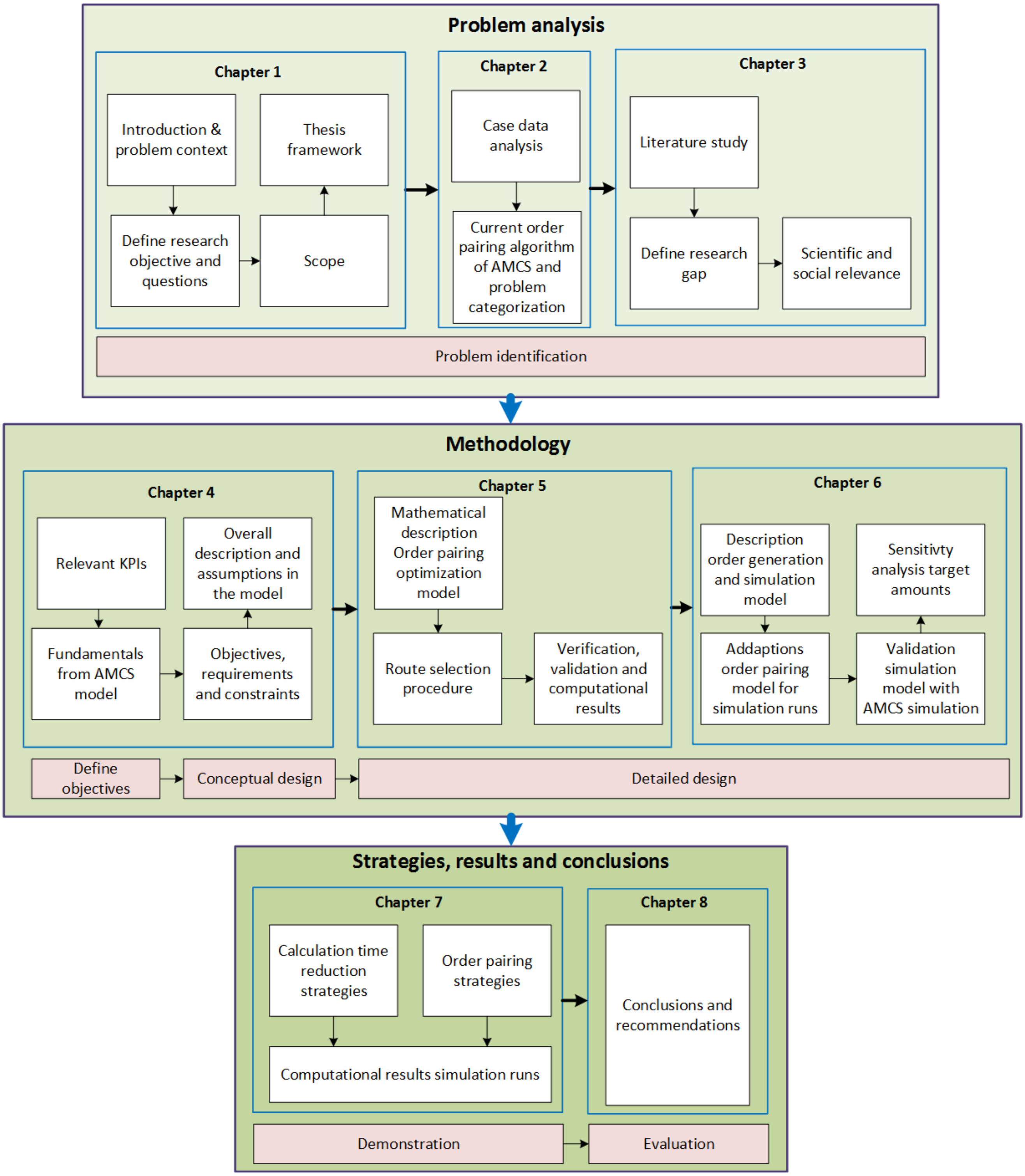

Figure 1.1: Framework used in this research.

2

Current order pairing strategy of AMCS

This chapter will analyze the current petrol station replenishment model used in the software of AMCS. Knowing more about the strategy that is used right now can help to understand the decision logic and identify what models from the literature have comparable features. As an introduction to this chapter, section 2.1 will describe the data that is used for their model, which is based on a specific case provided by AMCS. This same data will also be used in this thesis research. The strategy of AMCS actually uses two distinctive models, where the first one uses the input from the data it gets to generate orders which contain tank amounts and a delivery time window. This will be explained in section 2.2. The second model uses the output of the first model to try pairing orders in a single vehicle and assign routes to the vehicles, which will be explained in section 2.3.

2.1. Case study: data analysis

Before further going into detail about the order generation model, it is necessary to have an overview of the data that is used by AMCS as an input. The first part will give a general overview of the data, while sub–section two and three will go deeper into this by scaling the data down into stations and tanks and vehicles and compartments respectively. The last part will look into the variables that can be tweaked and the possible impacts of these variables on the model when changed.

2.1.1. The big picture

This section will give an overall picture of the data that is provided by AMCS. The data is derived from real–time measurements and therefore represents a real–life case. For this research the data will be anonymized. In total there are 59 stations served by one depot, with a fleet consisting out of 4 vehicles.

Two of the vehicles are semitrailer trucks (see figure 2.1), while the other two are rigid trucks which have the ability to take an extra trailer, also indicated as a drawbar trailer (see figure 2.2). The latter vehicle configuration allows for changes in the length of the vehicle, because the drawbar can be decoupled. This leaves the rigid truck which is smaller than the semi–trailer, leaving open the option to visit stations which only allow smaller trucks, for example in a city.

The stations allowing only certain vehicles have a qualification parameter matching the type of vehicle. Most stations are open 24 hours a day, 7 days per week. This allows flexible time windows and only the work schedules of the vehicles are of large influence. Another important feature following from the data is the presence of flow meters on the vehicles. This is not very common which already became clear while reviewing the literature. Knowing the vehicles are equipped with flow meters gives a lot more opportunities in the order pairing strategy.

Figure 2.1: Example semitrailer truck (HMK Bilcon, 2016a).

Figure 2.2: Example rigid truck with drawbar trailer (HMK Bilcon, 2016b).

The last thing that can be noticed in the data is that the start date of the measurements has started on a Saturday, Because of this the model will also have its start date fixed to a Saturday. The vehicles have different work schedules during the week and the weekend, which is something that has to be taken into account. The consumption rates are measured once a day, deriving an expected average consumption rate per tank from these values. The time interval for this in the data is from 00:00 till 23:59. This is not very important for the model, but it is something to be mentioned.

Now that an big picture is given on the data, the following sections will dive deeper into the data, analyzing which variables are given for the vehicles, stations, tanks and trips.

2.1.2. Stations and tanks

As mentioned before there are 59 station to be served in total. This will be done from one depot, so in total there are 60 locations for the vehicles to go. The locations from the data have a name and city, an ID number and longitudinal and latitudinal coordinates. The longitudinal and and latitudinal coordinates also have an X and Y coordinate respectively. None of these are required in the model, because AMCS has also handed travel time and distance matrices. Since these are determined directly from their own geocoding and measured average travel times, it is useless to do double work.

Furthermore the data contains information about the opening hours of the stations and the vehicles that are qualified to visit the stations. The depot is open 24/7, so vehicles can depart and arrive at any moment. The stations also have a certain amount of tanks. Mostly three to four, with different capacities and consumption rates. They all have their own ID, which is similar to the station ID, plus an additional number simply indicating tank 1, 2, 3, etc. Furthermore it is given what type of fuel is dedicated to a specific tank. The data gives six different products for delivery to the stations. While there are actually three types of fuel, namely petrol 92, 95 and regular diesel, there are two different brands for sale. Every station has their own 'preference', so the different brands cannot be freely exchanged for one another.

There are also restrictions towards the minimum amount that has to be delivered in one order, which in this case is the same with 1000 liters for every tank. An empty level is given, indicating the lowest level that a tank may have. Getting below this level is not allowed, as this would result in a late delivery. Derived from the capacity and minimum level a minimum and maximum stock level are determined by AMCS. The maximum stock level is 96% of the total capacity of a tank, which is determined by keeping in mind that the goal is to have stock levels at customers as low as possible. The minimum stock level is the level at which an order must be executed and is a chosen value based on two parameters:

- The first one is the consumption rate of a tank. With one tank having a faster depletion rate than the other, it is required to anticipate to this by having a larger minimum stock level. If for example two tanks at one station have the same size but different consumption rates, the tank with the faster consumption rate does have a shorter lead time. This can be compensated by giving it a higher minimum stock level.

- The second parameter influencing the height of the minimum stock level is the distance of a station to the depot. It is not hard to imagine that a station nearby can have a higher minimum stock level at which an order has to be executed. After all, the lead times are much shorter than a station that is located farther away.

One of the most important variables from the data are the estimated consumption rates for every tank per 24 hours. This data is collected over the course of one week, which can help to make an forecast over a longer time period. This can be done by taking the same consumption rates for every day over multiple weeks. This can help the model to determine when an order can or must be executed. The variables derived from the AMCS data, that can be used for the model in this project are listed in figure 2.3.

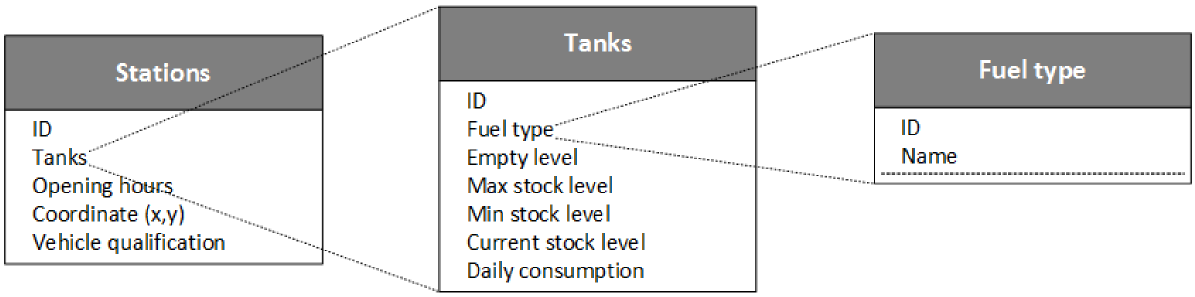

Figure 2.3: Important input parameters and relations between stations, tanks and fuel.

2.1.3. Vehicles and compartments

As stated in the first section of this chapter there are two types of vehicles: Rigid plus an potential extra trailer (rigid plus drawbar) and the semitrailer truck. All trucks have their own ID and the corresponding trailers have the same ID number, indicating that one trailer can only be used by one truck. Interesting is the fact that all trucks are equipped with a flow meter, which makes it possible to have different orders in one compartment.

In total four vehicles are available, serving all 59 stations. Two vehicles are semitrailer trucks with six compartments, having different capacities per trailer. The other two vehicles are rigid trucks with drawbar trailers, which have four compartments for both the rigid and drawbar respectively, thus eight compartments in total. The rigid truck and its drawbar have the same capacities for both vehicles. The data from the vehicles and trailers that have important characteristics for the model input are displayed in table 2.1 and table 2.2.

Table 2.1: Useful data about truck layout.

Trucks	ID	Total volume cap (l)	# of compartments	Compartment volumes (l)
Tractor	3706	0	0	NA
Tractor	3920	0	0	NA
Rigid	3921	33000	4	7000, 5000, 7000, 7000
Rigid	3923	33000	4	7000, 5000, 7000, 7000

Table 2.2: Useful data about trailer layout.

Trailers	ID	Total volume cap (l)	# of compartments	Compartment volumes (l)
Semi	3706	45000	6	10000, 10000, 7000, 5000, 6000, 7000
Semi	3920	53000	6	10000, 12000, 5000, 6000, 7000, 13000
Drawbar	3921	26000	4	7000, 5000, 7000, 7000
Drawbar	3923	26000	4	7000, 5000, 7000, 7000

Each vehicle has their own unique work schedule, which mostly depends on the day of the week. During a normal workday all vehicles are available for most of the time. During the weekend some of them are used less or not at all. This is due to the fact that most traffic is visiting station during workdays, due to people commuting between home and work. In the weekend commuting traffic is much less, leading to fewer orders during that time. The work schedules are shown in 2.3.

Table 2.3: Work schedules for the different trucks.

Vehicle ID	Saturday	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday
3706	06:00 - 18:00	06:00 - 18:00	06:00 - 18:00 18:00 - 04:00	06:00 - 18:00 18:00 - 04:00	06:00 - 18:00 18:00 - 04:00	06:00 - 18:00 18:00 - 04:00	06:00 - 18:00 18:00 - 04:00
3920		17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30
3921	05:00 - 15:30	17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30
3923	05:00 - 15:30	17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30	05:00 - 15:30 17:00 - 02:30

2.1.4. Possible impact of variables

Looking back at the data in the previous sections, there are several variables in this specific case that are fixed, while others are subject to changes. Obvious fixed variables are the station locations, the number of trucks and the size of the compartments of these trucks.

The variables that can be changed are mostly influenced by the decisions of the depot and the clients, whether or not influenced by laws and rules imposed by (local) governments. This is for example the case for the the opening hours of the petrol stations. While most of them are in operation full-time, some do have specific opening hours. This is probably contributed due to factors such as the location (rural versus urban) and the restrictions forced by municipalities in certain neighbourhoods.

The impact of opening hours largely depends on the amount of stations which actually have limits on this. If only two or three stations have restrictions the impact would be low, since it is easy to plan around these stations. But if a larger percentage of the total stations is having delivery hour restrictions, the planning would also have to deal with more conflicting orders. The combination of conflicting order times and the need for on-time delivery could lead to the necessity of more delivery trucks or broader schedules.

Another station related variable is the vehicle qualification. This is a restriction on the kind of vehicle that is allowed to visit a station and is mostly due to constraints in the truck size. A good example is a station in the middle of the city, which has less space for vehicles to maneuver. It would introduce a new kind of handling: decoupling of trucks. If only a rigid truck is allowed, it would mean that its semitrailer has to be left at the depot (or somewhere at a point along the route). This means that less capacity is available for that specific trip. So if more stations have those restrictions it will also increase the difficulty to assign vehicles and lower the overall daily capacity that can be transported.

As said before the impact of the opening hours and vehicle qualifications depends on the amount of stations that have these restrictions. If it is only a very small portion, it will probably have almost no impact at all. In this case it should be considered to be kept out of the model.

Other things that can be varied are the minimum and maximum stock levels of the station tanks. It has already been explained before how these boundaries are chosen. But since these boundaries are chosen on a safe margin to avoid depletion, it could be the case that the boundaries are actually to high. If this is the case, it could be a possibility to tweak this and see the impacts of these variables. Making the boundaries less restrictive could give extra options, although it must be kept in mind that distance and consumption rates have already been taken into account by AMCS, when they specified minimums and maximums.

Furthermore a valid order requires to have a minimum amount. This is kept the same for every station and is determined by the planner. This minimum constraint is necessary to create feasible orders. Falling below a certain amount will only lead to unnecessary high costs, because small amounts lead to more deliveries and thus more trips.

One thing that could possibly play a rather significant role is the fact that the depot has the ability to delivery three types of fuel from two different brands. Every station has its own brand and although both brands have the same specifications, it is not allowed to blend them for obvious reasons. This restricts the use of multiple orders per compartment, making it more difficult to use order pairing. This is an important consideration for the used model and it might be considered to use three types of fuel instead, without looking at the brand or only allow order pairing for stations with the same fuel brand.

The last variable is the work schedule of the vehicles. This is another factor which has a large influence on the system. Ideally a vehicle is available 24/7, since orders can be planned whenever the vehicle is available. however, this is not realistic when looking at factors like working hours of drivers and maintenance. In this specific case the out of service hours are at most four hours during a normal weekday. During the weekend the work schedules are shorter and one vehicle is not used at all on Saturday. If all vehicles are operational 24 hours per day, it is easy to place routes in the schedule as only the time windows of orders have influence on where they are placed within the schedule. But when the schedules are split in a morning and afternoon schedule, the model has to search in both these slots and still has to take into account the time windows of the orders. This gives a lot more difficulties and can increase calculation times.

2.1.5. Conclusions on the data

This first section gave answer to the the first sub question: *"How does the case data provided by AMCS look like and how can it be used in the model?"*

The given case contains 59 stations that are served by one depot, which uses four heterogeneous trucks. Within the system there are two important types of objects available: stations and trucks. The station (and depot) locations determine distances and travel times, while they also hold the information about stock levels of tanks, which in turn lead to the generation of orders. The trucks are the objects moving orders from the depot to one or more stations, and their size and configuration determines which order they can handle. Because every tank at a station has their own unique characteristics (e.g. consumption rate, volume, etc.), tanks at stations can be seen as separate objects. Because of this tanks are seen as separate bodies within the stations. Table 2.4 gives an overview of the different objects with their input variables necessary to use in the model.

Table 2.4: Overview of input variables for the model.

Classification	**Input variables**
Stations	Tanks; opening hours; coordinate; vehicle qualification.
Tanks	Fuel type; daily consumption; min/max stock level; current stock level.
Trucks	Truck type; work schedules; number of compartments; compartment volumes.

The variables that are expected to have the most impact on the output are the work schedules of vehicles and the amount of different fuels available. The first one determines how much hours a truck is available during the day and has a big influence on the amount of orders that one vehicle can deliver during the day. The latter input, the variety of fuels available, can have a large impact when different orders are combined in the same compartment. The opening hours and vehicle qualifications of stations will have an increasing impact when more stations do have restrictions on this. The minimum and maximums of stock levels and order volumes can also have a fairly large impact when changed. They could influence the amount of orders, the amount of kilometers driven and the way that orders are paired.

2.2. Current order generation strategy of AMCS

This section will elaborate on the way that orders are generated in the current situation. The first subsection will describe the inputs that are used for making new orders and the second part will explain how orders are generated based on these inputs.

2.2.1. Inputs used in the model

Orders have certain information stored into them which is taken from the data explained in the previous section. This includes the station which needs a delivery, the amounts per tank of that same station and the time window in which delivery should take place. Orders in turn need certain information to be able to actually make them. It starts with adding the information about the vehicles, stations and its corresponding tanks to the model. These three types of 'objects' have the following predefined characteristics:

Stations – For the stations it has to be known which vehicles are allowed to visit them. Some stations have restrictions due to lack of space, making it impossible for some larger vehicles to access the station. Furthermore some stations have specific opening hours. While nowadays it is quite normal for most stations to be open 24/7, some may have restrictions on the hours that deliveries can take place. Finally the travel times and distances between the stations (and depot) have to be known, since this helps to determine if certain combinations of stations on a route are feasible.

Tanks – The stations have multiple underground tanks to store fuel. The tanks have their own set of unique parameters such that they are approached as separate entities within the stations. Obviously every tank has their own unique fuel type, which cannot change. All tanks have a minimum and maximum stock level. The maximum is determined as a percentage of the capacity of a tank. The minimum stock level is determined as a certain amount above the *empty level*. The empty level is the amount for which a tank is specified as depleted. The minimum stock level is the threshold to place an order in the system, to have enough time for a delivery to take place.

Vehicles – Trucks are of a specific type, semitrailer or rigid/drawbar, all having their own compartment configuration. This can mean a different amount of compartments as well as different volumes per compartment. This means the fleet of vehicles is of heterogeneous composition. Furthermore the trucks have their own work schedules. This weekly schedule is giving the daily time slots at which the vehicle is allowed to be used. For example, a vehicle on a normal working day can have a schedule from 05:00 until 15:30 and then from 17:00 until 2:30, having four hours in total in which the vehicle is on a 'break' and thus cannot be deployed.

Apart from the constants given above the model needs to have input about the actual tank levels. The stock levels of every tank are measured once per day at 00:00 hour. Based on this information orders are generated when one of the tanks reaches the predetermined minimum stock level. This means that new orders are not issued by the stations, but the depot itself decides when an order needs to be made and executed. When orders are made at the beginning of the day, the algorithm does assume there are no other orders planned in the future. This means that if an order was already assigned the previous day, but did not take place yet, it will be seen as if it was never created.

2.2.2. Generating orders and time windows in the current situation

With the inputs received by the model, the order generation done by AMCS is divided into three steps as follows:

1. When measuring the stock level at midnight it is determined which tanks have reached minimum stock level. The corresponding station to that tank then gets labeled with an expected depletion time similar to that of the specific tank. To put it in short: the earliest expected depletion time of a certain station depends on the first tank at that station to be empty. It is not allowed to deliver orders later than the earliest expected depletion time of a station. For every station the latest possible delivery time is set based on the first tank that gets depleted. The actual delivery time is then derived by finding one that is both feasible as well as practical. This is again restricted by certain rules:

   a. The delivery time should be as close to the latest possible opening hour of the station as possible, but always stay before the earliest expected depletion time of that same station.

   b. The order should fit on at least one compatible vehicle and the delivery time should fit in the vehicle's operating hours.

   c. Since there is some service time at the stations, e.g. the time it takes to transfer fuel from the truck to the station the delivery time should have a clear margin to deliver the order before depletion time.

   d. The time window should not be to short. If the expected depletion time is happening 2 minutes into the opening hours of a station, the time window would be to narrow to use this delivery time. If this is the case the delivery time should be shifted towards the previous opening period instead.

   e. Taking the above into account, the differences in operating hours of vehicles, opening hours of stations and depletion times should be wide enough to allow a vehicle to have enough margin on the driving time to get from the depot to a station and back to the depot. If this time window is to small, the order should be delivered in the previous period instead.

   f. If all of the above is taken into account but it is still impossible to find a feasible time window, the last resort is to look at opening hours later than depletion time. This last resort rule means depletion will happen and thereforee should be avoided at all costs.

2. An order must be fully delivered by one vehicle and cannot be split over several trucks. Moreover the orders are made per station, because it is not allowed to have more than one vehicle per day visiting a station. This means that an order has an amount for every tank at the corresponding station. The size of every order is determined by looking at the expected tank levels at a station at delivery time. When having a heterogeneous vehicle fleet it is impossible to fix orders to compartments right away, because a vehicle with the right size is not always available. thereforee order generation is working with target amounts and optional amounts, which is decided through the following rules:

    a. A *target amount* is the amount for every tank at a station which will push the depletion time ahead to a new desirable time. Considering the one order per station rule, the order for one station contains the target amounts of every different tank located at that station. It is tried to reach target amounts that would ideally lead to a situation where all tanks reach depletion at the same time (e.g. simultaneous dry run). The target amounts are then defined as the smallest amounts that would push the stations expected time of depletion as far as possible into the future if delivered at the latest possible delivery time. This could also mean that some low consumption tanks might not need anything delivered at all because they can survive for much longer with what they already have.

    b. The *optional amounts* are maximum volumes on top of the target amount that would not lead to overflow of the station's tanks.

    c. The target amounts cannot be higher than the carrying capacity of the largest truck in the fleet.

    d. *Optional amounts* are only limited by the individual tank's capacity.

    e. There are also *minimum amounts* possible. This is possible with a cut down on the target amount defined as the maximum percentage for which an order may deviate from the intended amount. This cut down is typically between 10 or 20 percent.

3. At last an actual time window is generated. This time window is found by looking at the latest opening time of the station and set the window to this. If it would end after the latest delivery time in this situation, it is set back to this time instead. In the case that a time window begins while the minimum amount would violate the tank capacities, the time window is shifted towards the earliest time that is feasible towards the tank capacities. Furthermore the time window can be extended by moving its starting or ending time if this is necessary to fit the minimum duration of an order. Generating a time window is further restricted by the following rules:

    a. Obviously the time window cannot take place in the past and should not start earlier than *time–now*.

    b. The time window is large enough to be able to fit travel times and service times into it.

    c. A vehicle just returning from a trip needs some preparation for the next trip, so the first few hours after *time–now* do not count towards the lower limit of the time window.

    d. Multiple time windows should be considered.

    e. The time window does take into account the opening hours and operating hours of the stations and trucks respectively.

2.2.3. Summarizing the order generation algorithm

To briefly summarize the above, the following steps are necessary to generate an order:

1. Measure the stock levels at the beginning of each day. When one of the tanks at a station reached minimums, an expected depletion time is made and a first delivery time window for that specific tank is made.

2. Because it is not allowed to have more than one vehicle per day visiting a station and an order must be fully delivered by that one vehicle, an order has an amount for every tank at the corresponding station. The size of every order is determined by looking at the expected tank levels at a station at delivery time, with the goal to push next depletion as far as possible. This are *target amounts* for every tank. If allowed by the compartments and tank capacities *additional amounts* can be added. Next to that it is possible to have *minimum amounts* if a wider time window is needed.

3. The last step is to make the actual time window. Which should be large enough to take into account travel times and service times at the station, while it should also respect the opening hour of stations and the working schedules of trucks. Furthermore it must also be taken into account that the vehicle should always start and end at the depot.

After this an order is generated for a station, with the following information:

- Station ID.
- The time window $[T_a, T_b]$ in which delivery can take place.
- T_a is the latest possible delivery time based on the tank that is depleted first.
- T_b is based on the minimum amounts that do not violate tank capacities.
- minimum, target and optional amounts for every tank at the station. Where the amounts are varying based on the final delivery time that gets chosen.

2.3. Combining orders in the current situation

With the information about the generated orders it can be considered to pair certain orders. This is done by considering the orders with matching time windows and taking the the involved stations as one order. Then the new order amounts are generated and taken as one order with a two new time windows. The delivery times do not have to be the same, because a truck cannot be on two places at the same time.

To help the algorithm in the order pairing, certain help is handed to the software. AMCS has used station grouping where orders of the same group are allowed to be paired, but no pairing is allowed between different groups of stations. Furthermore only orders that are not considered full are taken into account, where all orders above 90% of the truck capacity are labeled as full. So the algorithm is asked to look at non–full orders in the same station group.

All the non–full orders are then paired based on the earliest delivery times. This results in an order that most often is full, but some order pairs could even use a third or fourth order. This depends on the upper limit of orders that is specified by the user to the program. An example would be to have two orders per pairing, where the first two would have the earliest delivery times, the second order would have the third and fourth earliest delivery times, etc.

The 'slow stations', which have low consumption rates, do not have to have as much deliveries as stations with high consumption rates. But since the algorithm sees paired orders as one, it would only anticipate to have one depletion time for both stations next time. This means that both stations will be depleting at the same moment leading to the same paired order next time. The problem with the slow stations is that they can probably be visited a lot less than by just giving them the target amounts specified. thereforee the slow stations are labeled as 'prioritized' meaning that they should always have their target amounts when paired with another order.

The opposite of prioritized stations is also possible. These stations, labeled 'leftover stations', have less priority and take up any volumes that might be left in the trucks. The spare volumes going to these leftover stations do not replace the original order and it is possible that several deliveries take place to the leftover station. Most of the times this involves larger stations that have high consumption making it easy to deliver spare volumes of the vehicles to this station. Furthermore this station does not have to belong to a certain group of stations when paired.

When the order is planned it has to be assigned to the compartments of a truck. When orders are divided over compartments it will first assign the minimum amount. While trying this for as much orders as possible to get more order pairs. After that the algorithm will try to increase the volume of the tank with the highest priority and if all tanks have reached target amounts, it can finally try to add optional amounts to fill up the vehicle.

2.4. Overview of the AMCS models and problem categorization

Section 2.2 and 2.3 answered the sub question *"How does the current order pairing algorithm of AMCS look like?"*.

It became clear that the model can actually be divided into two sub models: An order generation model and a model that tries to pair orders and assign them to vehicles, as shown in figure 2.4. More detailed information about the inputs shown in the figure are given in table 2.5 This structure is used as a reference for the design of the model in this research.

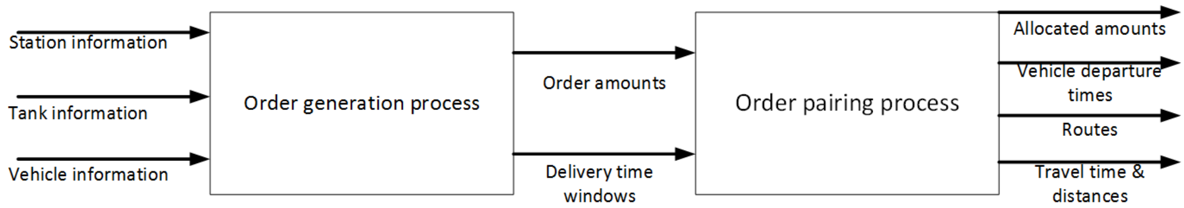

Figure 2.4: Overview of the order generation and order pairing model, with the flow of input and outputs.

Table 2.5: Inputs for the order generation model.

Model inputs
Station information: opening hours, coordinates, vehicle restrictions
Tank information: current stock, daily consumption, minimum levels allowed, fuel types
Vehicle information: working schedules, compartment sizes, type

Because the order generation model is an algorithm that can create orders based on certain elements like forecasting, it is rather simple to construct. The order pairing model is a lot more complex and can be categorized as a Petrol Station Replenishment Problem (PSRP), which is a special case of the the conventional vehicle routing problem (VRP) and could also be identified as a capacitated vehicle routing problem (CVRP) or inventory routing problem (IRP) (Popović et al., 2011; Li et al., 2014). These problems have vehicles with a limited carrying capacity for the goods that have to be delivered. This has led to the combination of two traditionally separated areas, concerning the routes of vehicles and the loading of the goods on those vehicles (Iori & Martello, 2010). With both separated problems considered NP–hard, combining these areas increases the difficulty of solving them (Iori & Martello, 2010). Solving large scale problems by using exact methods are quickly becoming impossible due to large computing times. With a few dozen stations all having their own tanks and consumption rates, many solutions are possible and finding a solution can take up a lot of calculation time.

Table 2.6 gives a short overview of the distinctive characteristics that belong to the specific PSRP of AMCS:

Table 2.6: Overview of the order pairing model characteristics of AMCS.

Model characteristics
Objective: Minimize travel distance and stock at stations
One depot
Limited heterogeneous fleet
Station vehicle restrictions
Consumption rates measured once per day
Vehicle working schedules
Station opening hours
Time windows
Minimum, target and optional amounts.

The next chapter will look into the literature relevant to the PSRP problem and compare this to the current order generation strategy of AMCS.

3
Literature study

Research in vehicle routing problems (VRP) has sharply risen since the beginning of the information age. Dating back from the first algorithm introduced in 1959 by Dantzig and Ramser the ever increasing power of computers has led to the development of hundreds of algorithms and models (Barman, Lindroth, & Strömberg, 2015). This chapter will look into the literature involved with the delivery of fuel to petrol stations using trucks and try to find similarities with the order pairing strategy of AMCS.

3.1. Literature on the petrol station replenishment problem

A special case of the VRP is known to be the capacitated vehicle routing problem (CVRP). A systematic overview of the different kind of problems involving CVRP which have been researched are given by Iori and Martello (2010). One of those variants on the CVRP is the petrol station replenishment problem (PSRP). This routing problem aims to optimize the delivery of petroleum products from a depot to several petrol stations over a certain planning horizon, minimizing travel costs (Cornillier, Boctor, Laporte, & Renaud, 2008a). In this specific problem not only the routing of vehicles is important, but also the optimal filling of the goods inside the vehicles. This is especially the case when having a heterogeneous fleet of trucks with varying carrying capacity. On top of that it is possible to have a different setup of compartments on vehicles with similar carrying capacity (for reference see figure 3.1).

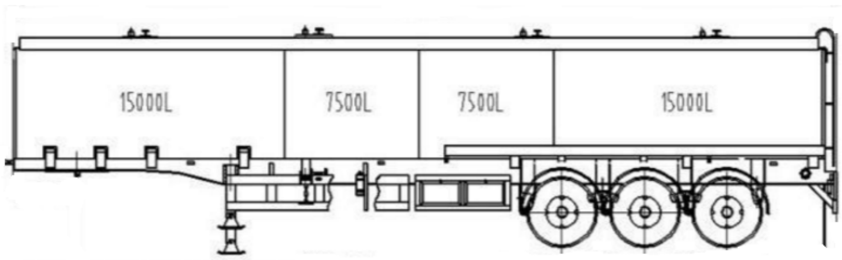

Figure 3.1: Typical tanker truck with different sized compartments. Adapted from Shandong Liangshan Tongya Automobile Co., Ltd. (2016).

For example, two trucks with an identical carrying capacity of 40.000 liters can differ in tank setup such that one has four tanks of 10.000 liters, while the other has five tanks of 8.000 liters. Other deviations from a classical VRP are that the compartments can carry only one specific fluid and the trucks are lacking flow meters, making it impossible to divide the volume of one compartment over multiple stations (Toth & Vigo, 2002).

When considering a heterogeneous tanker fleet with different tank capacities and having multiple orders to take care of, the delivering company must choose the right vehicle for the right destination. It is undesirable to send the truck with the largest capacity to a gas station that only requires 20% of the volume this truck can carry. Although this is considered undesirable, it is not unlikely that it will occur once in a while. One of the findings from a study carried out by Avella, Boccia, and Sforza (2004) concluded that the placing of orders depends on many factors, such as climate and season. Because of these uncertainties the frequency of orders can heavily vary over time. therefore it is impossible to have a fixed delivery schedule to clients and, since most delivery companies have a policy to deliver withing 24 hours, it is sometimes necessary to dispatch a vehicle which normally would not be used for that particular order.

In the model of (Avella et al., 2004) it is assumed that the different compartments of truck will either be empty or full; a specific fuel of a particular order cannot be split over multiple compartments; and the content of one compartment will be delivered to one client. The allocation of orders to the vehicles is based on *Best Fit Decreasing (BFD)* (Simchi-Levi, 1994; Clarke & Wright, 1964).

Cornillier et al. (2008a) has analysed the same sort of problem, only differing in the conception that compartments and order sizes do not have to match completely, such that orders can be split over multiple compartments if necessary. The only constraint is that a compartment must be completely empty before it is filled for another order, making it impossible to combine orders in one compartment. Another constraint that was implemented involved a maximum amount of two destinations per trip, as it is expected by Cornillier et al. that the destinations per truck rarely exceed two per trip due to the capacity of one truck being to small to serve more. The model also involved an heuristic algorithm to have procedures for anticipation or postponement of orders.

The research by Cornillier et al. (2008a) was further extended over the following years. Additionally Boctor, Renaud, and Cornillier (2011) did research towards the trip selection procedure, which arose as a sub–problem from the first model. Contrary to the first model, which used an unlimited amount of heterogeneous vehicles, the other problems did have a limited heterogeneous fleet. Additions to the model included multi–period analysis on a real–world case (Cornillier, Boctor, Laporte, & Renaud, 2008b), using time windows (Cornillier, Laporte, Boctor, & Renaud, 2009) and having multiple fuel depots instead of only one (Cornillier, Boctor, & Renaud, 2012).

For the latter two problems the model was extended to be able to determine trips with three or four stations instead of the earlier two station constraint. According to Cornillier et al. (2009) most vehicles have four to six separate compartments, while stations require at most two or three different products. Considering that for one kind of fuel at most two compartments will be delivered to the same station, there is a possibility that trucks can visit more than two stations. A remark made by Cornillier et al. (2012) concerning the multi–depot problem was that it is hard to solve, taking a lot of computational time and even can become impossible. This is why they used a trip selection model, which selects a trip subset out of a candidate trip set with the aim to maximize the net daily revenue.

Popović et al. (2011) adds inventory management in which the supplier determines how much and when fuel is delivered to customers, instead of customers randomly sending order themselves. This way the objective is not only to focus on lowering the vehicle routing costs, but also lowering inventory costs for the oil company. Popović, Vidović, and Radivojević (2012) state that fuel consumption is a stochastic process instead of a deterministic one. This is why they introduced a simulation approach with a deterministic IRP and a stochastic fuel consumption. Furthermore Popović et al. (2011) makes use of Variable Neighborhood Search (VNS), first introduced by Mladenović and Hansen (1997). Differing from the common Tabu search (Glover, 1989), this search algorithm uses the current most optimal solution and searches in the neighborhood for better solutions, increasing the distance of neighborhoods when no better solution is found. If a better solution is found however, the algorithm will use this as the new best optimal solution, repeating the search process. This method ensures that certain favorable characteristics of a solution are kept and used to find better optimal solutions (Mladenović & Hansen, 1997).

Elaborating on the same problem (Popović et al., 2012) introduced two other approaches for solving it. The MIP model looks for stations to assign to certain routes taking into account (daily) inventory costs. An additional heuristic introduced a relaxed MIP making a first initial solution, after which the heuristic will search for possible deliveries that could be delivered in an earlier time slot and look if other stations can be added to the current route.

Li et al. (2014) mentioned that out of stock avoidance is sometimes more important than lowering transportation costs. Taking this into account, their research puts focus on minimizing the maximum route travel time rather than aiming for lower costs. The can do and must do orders are decided by specifying lower and upper bounds for the delivery times, where the latest time for delivery is out of stock time and the earliest delivery time is depending on a minimum stated delivery quantity to maintain an adequate utilization of the transportation resources (Campbell & Savelsbergh, 2004).
More recent research on the PSRP mostly elaborate on the earlier implied meta heuristics and has been focused on real–life problems, giving them their own set of properties and constraints for that problem. Benantar et al. (2016) looks at a comparable problem to Cornillier et al. (2009) but adds a constraint that some petrol stations cannot be visited by certain trucks (in their case private owned vehicles cannot visit army fuel stations).

Coelho and Laporte (2015) give four classifications for the delivery of fuel orders, which depend on two things: order splitting over multiple compartments and order splitting over multiple vehicles. Coelho and Laporte rightfully state that most problems in the literature are using the unsplit–unsplit classification. Most research is conducted using this classification due to two reasons mentioned before in this literature study: 1) Traditionally trucks are not equipped with flow meters, making it impossible to split a compartment and 2) Splitting an order over multiple vehicles is not desirable because of time windows given by customers.

Coelho and Laporte are questioning these reasons stating that flow meters can be installed if necessary. Also, the decision to deliver an order with multiple vehicles is a purely economical consideration if customers allow multiple trucks in a certain time window. They modeled multiple routing problems covering all four classifications. The success of the models were mostly depending on the use of a single– or multi–period instances, where multi–period proved far more challenging to solve. The latter has been confirmed by Triki and Al-Hinai (2015) who has done additional research to an extended planning horizon (larger than one day) and states that while a longer planning horizon could lead to more cost–saving, it also "increases the complexity in the underlying optimization problems to be solved" (Triki & Al-Hinai, 2015). In other words, stretching the planning horizon gives more decision variables to take in to account, which makes it more complex to solve and will significantly increase calculation times.

3.2. Conclusions on the literature

while the order pairing strategies for the PSRP differ per paper, most do not allow a vehicle compartment to hold multiple orders on one trip because they assume trucks do not have flow meters, making it impossible to have multiple orders in one compartment. With this assumption most vehicles have enough compartments to supply two stations on one trip, but three or more stations get significantly more difficult, since most trucks do not have more than 6 to 8 compartments. Another reason for the one–order–per–compartment rule is that it significantly increases computational times, since it gives much more possibilities for bin packing. The order pairing algorithms in most models are using greedy heuristics like first fit and best fit decreasing. Additionally to the compartment constraint, most models have the constraint that an order from one station should always be delivered by one vehicle. This means it is not allowed to have split orders over multiple vehicles, introduced because of the idea that a station is only allowed to be visited by one truck per day.

The objective functions used in the papers are mostly about keeping costs as low as possible. The way this is done does vary a lot, where some papers only induce costs on travel time or distances, while others have a more detailed picture, like including inventory costs and penalties when a certain vehicle is visiting a specific station with vehicle type restrictions. Most articles define that petrol stations should always have enough fuel to serve its customers, so when the level of one of their products is getting low these should be refilled before depletion. For some models this is a hard constraint, while others give penalty costs when a tank reaches minimums.

3.3. Relations between the literature and order pairing model of AMCS

To aswer the third sub question, *"What comparable characteristics and models are available in the literature?"*, this section will look for comparisons in the algorithm of AMCS and the available literature. The order generation algorithm of AMCS has certain specific characteristics. First of all, time windows are made based on the depletion times, truck working schedules and opening hours of the stations. If no feasible time window is found to be available near depletion time, the algorithm will look for an time window in an earlier period. Since it is not allowed to have depleted tanks, the latest delivery time of an order is determined by the tank that will first go empty.

The literature that includes time windows is limited to three recent studies: Cornillier et al. (2009),Li et al. (2014), Benantar et al. (2016). Cornillier et al. uses time windows based on working hours of employees, with possible overtimes. They do not take into account opening hours. Both Li et al. and Benantar et al. do use working hours of vehicles without overtime and no opening hours are considered. The latter two specify the earliest possible delivery time as a minimum amount that has to be delivered to make an order. For the latest delivery time all three problems specify the depletion of the most critical tank. None of the found literature is basing its time windows on the opening hours of stations.

Furthermore AMCS uses minimum, target and optional amounts. Just like in this research, most literature uses deterministic consumption rates taken from historical data. Amounts are specified by using forecasting to see when tanks will go empty. If no feasible time window can be found when using target amounts, the model can choose to go to minimum amounts instead. The same sort of algorithm is used by Benantar et al.. Li et al. uses the same sort of rule, but it is limited because tanks at stations should always be filled to the maximum capacity at delivery time. Both studies are not working with optional amounts, but this might be added as a new rule being an 'in between' option of both studies.

AMCS also states that one order should be fully delivered by one truck, but a truck can deliver multiple orders on one trip. If the trucks have flow meters, they have the possibility to take multiple orders in one compartment, as long as it is the same fuel type. While most literature uses the one order one truck constraint and restricts to have one compartment only holding the order of one tank, Coelho and Laporte looked into flow meters, making it possible to allow use split compartments.

The fact that the current problem has some stations with limitations to the kind of trucks that are allowed is fairly unique in this field. The only study doing a similar kind of thing comes from Benantar et al., which makes distinction between depot owned and private owned trucks. Penalties are induced when a private owned vehicle has to be hired. So this is tried to be avoided at all costs. Next to that only army vehicles are allowed to visit army fuel depots. In the case of AMCS some stations do not have space for larger trucks with a trailer. So this is something that can partly be implemented from Benantar et al..

When looking at the objectives of the model, avoiding depletion of tanks and trying to minimize the mileage, only Triki (2013), Li et al. (2014) and Vidović, Popović, and Ratković (2014) have comparable objectives. Most literature focuses on costs rather than driven kilometers. However, the difference between distances and costs are based on the data provided. Since AMCS has not provided data about costs, it is impossible to aim for minimized costs.

3.4. Time period and planning horizon

In the problem description it was stated that AMCS wants to look over a longer time horizon than one day, because extended periods could have a large impact on the way orders are paired, whereas the algorithm could delay or advance certain orders in time when it will lead to improvement to the overall schedule and lead to delivery savings. This is backed by multiple studies such as Benantar et al. (2016), Coelho and Laporte (2015), Vidović et al. (2014), Triki (2013), Popović et al. (2012), Popović et al. (2011) and Cornillier et al. (2008b).

Most research took a time period of one day with multiple periods behind each other. This means the algorithm plans the specific day it is currently starting, while not taking into account tank levels for future days. This leads to less effective planning, because most stations do not have to have a delivery every day. Making the time period larger, for example three days, would give more space to form better order pairs. Currently the only problem that analyzed a time period longer than one day is Cornillier et al. (2008b) and since the problem was to complicated to use optimal solutions, it implemented an heuristic which has found to be promising enough to be used for other cases.

Another more recent algorithm and four corresponding heuristics have been proposed by Triki. Instead of making routes and order first, the idea was to look when stations have feasible slots and how these can be combined with delivery windows of other stations. So first all deliveries per day are made over the whole time horizon, after which routes are chosen. Two of the heuristics showed promising results, but still are unusable when looking for a longer time period than a week.

3.5. Research gap found in literature

Together with the previous chapter which analyzed the current algorithm of AMCS, this chapter has identified which models in the papers have similar characteristics as the one AMCS is using.

By comparing the characteristics of the AMCS model with the cases described in the literature some important conclusions can be drawn. AMCS has used combinations in their algorithm that have not been done by any literature in the same way. The closest examples are Cornillier et al. (2008b), Cornillier et al. (2009), Triki (2013), Li et al. (2014) and Benantar et al. (2016). However, most of these studies have done only one or two aspects that are found in the AMCS algorithm, but never the specific combination of station restrictions, time windows, vehicle working schedules and a heterogeneous limited fleet. Furthermore most literature is only focusing on optimizing the PSRP problem for a single–day period, whereas the research in this thesis is aiming to look for a longer time–horizon. This shows that there is a research gap found between the available literature and this specific case problem.

The models from Benantar et al. and Coelho and Laporte seem to be the most promising, since these have good comparisons with the AMCS model. This could be a good basis to start from when assessing the mathematical model. The other models mentioned before are possibly good additions to the two other models.

3.6. Scientific and societal relevance

This research is relevant from a social as well as an academic perspective. The academical relevance of this research focuses on the knowledge gap that is apparent. The social relevance focuses on the contribution of this project to society. Both will be briefly discussed in this section.

3.6.1. Academic relevance of this research

As mentioned in the introduction the call for more efficient transport has led to an increasing attention towards planning and routing optimization for this problem. While the routing and packing problem has been around for several decades, the PSRP is still fairly new and the improvements are mostly from the last 8 to 10 years.

The literature has addressed some combinations and issues already, but never the specific combination of station restrictions, time windows and a heterogeneous limited fleet. This research will aim to do just that. Generate new strategies by using the elements from the literature in a unique combination. It will implement and test the strategies on real–life data. Furthermore it will differ from most literature in the sense that travel distances and on-time delivery are the most important factors, whereas most research only aims to reduce costs. Due to the unique combination, the results of this research could potentially lead to new insights on how to approach the PSRP.

3.6.2. Economic and societal relevance of this research

As has been outlined in the introduction, transport is an important factor within the supply chain of a petroleum company. While oil gets more and more valuable due to increasing demand and a decreasing amount of sources, it is inevitable that prices will increase throughout the years. Transportation is one of the things that could be improved to cut costs and prices at the petrol stations. With transportation costs representing up to 25% of the final product costs (Avella et al., 2004), research and development of VRPs have proven to be relevant for the transportation sector and have had success in terms of saving costs, ranging between 5% and 25% of the total travel costs (Ballou, 2004; Avella et al., 2004).

It is also a relevant topic because of the increasing attention towards efficient transport from governments and environmental organizations, as well as the transport companies themselves. If improvements to the order pairing strategy lead to less driven kilometers as a whole, this will naturally lead to less carbon–dioxide emissions, lowering the carbon footprint. This is an important incentive for AMCS to deliver more efficient software.

The model introduced in this thesis can be a first tool to try and generate new strategies in the order pairing. Tests that lead to promising results could possibly be implemented in their software. The next chapter will look at the goals of AMCS, which helps to define the most important KPIs. After that the model requirements, objectives and constraints are specified, from which a conceptual model is proposed.

II

Methodology

4
Conceptual model

The literature study and the overview of the current order pairing strategy in the previous chapters have led to a clear research gap. The design of a model having the unique combination of time windows, vehicle restrictions, a heterogeneous limited fleet and a multi–period simulation has not been done before. This chapter will explore how the conceptual design of the model should look like. The first part of this chapter will explore the goals and key performance indicators (KPIs) important for AMCS, which will serve as the most important output of the model. The first section will look into the goals of AMCS, as to why they want to improve their order pairing strategy. The second section will further define the chosen KPIs and their respective units of measurement. After that some supporting output variables will be defined, which serve as an aid to check if the model is behaving as intended. Section 4.5 will elaborate on the fundamentals taken from the AMCS case which will also be integrated into this model. After that the requirements, objectives and constraints for the model are stated in section 4.6. The last part will give a formal description of the model and state the assumptions used.

4.1. Defining the goals of AMCS

As stated in the problem description AMCS want to improve their planning software with the goal to provide petroleum companies with a tool that will lead to more efficient and environmental friendly transport. With this in mind as a reference, while consulting AMCS it became clear that the following factors could be defined in decreasing order of importance:

- on–time delivery (before a tank at a station gets depleted).
- Avoiding stock buildup at the petrol stations.
- Reducing mileage.
- reduce CO2 footprint.

The first goal is the most important because it is a certainty you provide as a petroleum company towards your clients, the owners of the petrol stations. It is usually not acceptable that deliveries are too late, since that would lead to serious problems in the fuel supply to road traffic. This in turn could have economical consequences for society, of which the recent fuel strikes in France are a good example (The Local France, 2016). Most petrol stations have demanded a contractual liability if a late delivery might happen. Late deliveries would thus harm the petroleum company in reputation and financially.

The second goal, avoiding stock buildup, is desirable because having (almost) fully filled tanks at the stations means that valuable volumes of excess fuel essentially become wasted assets that cannot be invested elsewhere. To put it simply, excess fuel in tanks could potentially have been allocated to an extra station, increasing the total sales.

The third goal is important for the petrol company itself and has to do with cost reduction. Not only could lowering the costs potentially lead to more profit, it could also lead to a more competitive business where delivery prices can be lowered, indirectly improving the relationship with customers. Not taking into account fixed costs, the operation costs of trucks are depending on their utilization. It is desirable to keep the this as low as possible, while still being able to deliver to all stations. Sometimes having a truck making multiple stops on a route is feasible, while at other times it is not. This depends on the amounts of fuel delivered, as well as the additional travel time required for an extra stop and the size of the truck that is used. However, when referring back to the on–time delivery constraint, if a certain route is not deemed feasible it does not mean it should be skipped if that would lead to a late delivery otherwise.

On the other hand it might be thinkable to deliver a bigger or extra order to several stations if excess truck capacity allows to do so. This could be wanted when an extra order improves the order allocation for the next day(s). But it must be kept in mind that in most cases it is more viable to have two trucks on full capacity to visit one station each, rather than visiting two stations per truck, as this would lead to an extra trip between two separate stations.

The first three goals together with the chosen strategy can have a great impact on the utilization of trucks. Compartments for example can be filled with multiple orders when flow meters are used. This could lead to higher utilization of compartments, which possibly leads to less trips. AMCS mentioned that they did their own simulations, which sometimes generated orders that had the largest truck going to the furthest station with only 20% of compartments filled. Naturally this is undesirable and should be avoided when possible, even though it is probably impossible to avoid this for every situation.

The last goal is to limit the CO2 emissions by trying to have more efficient transport. This could include lowering the amount of driven kilometers or have more efficient vehicles. The latter one is not falling inside the scope of this research and will not be highlighted any further. Lowering the amount of driven kilometers is a means to lower the CO2 emissions and has a linear relation.

Based on the information given above, the on–time delivery is considered so important that this will not be approached as a key Performance Indicator, but it will be translated into a a hard constraint of the model. Thus the model is constrained to make orders that will always lead to on–time delivery. The other three goals can be translated into KPIs, which will be discussed in the next section.

4.2. Key Performance Indicators related to goals

To evaluate the performance of alternative order pairing strategies it is required to have some indicators that translate the goals of AMCS into measurable components. This section will discuss the KPIs that are important for AMCS. With the main goal discussed in the introduction and having defined which sub goals are important for AMCS, it is possible to scale this down into measurable variables. Figure 4.1 gives an overview of the goals and the criteria belonging to these goals. All KPIs will now be shortly described in separate subsections.

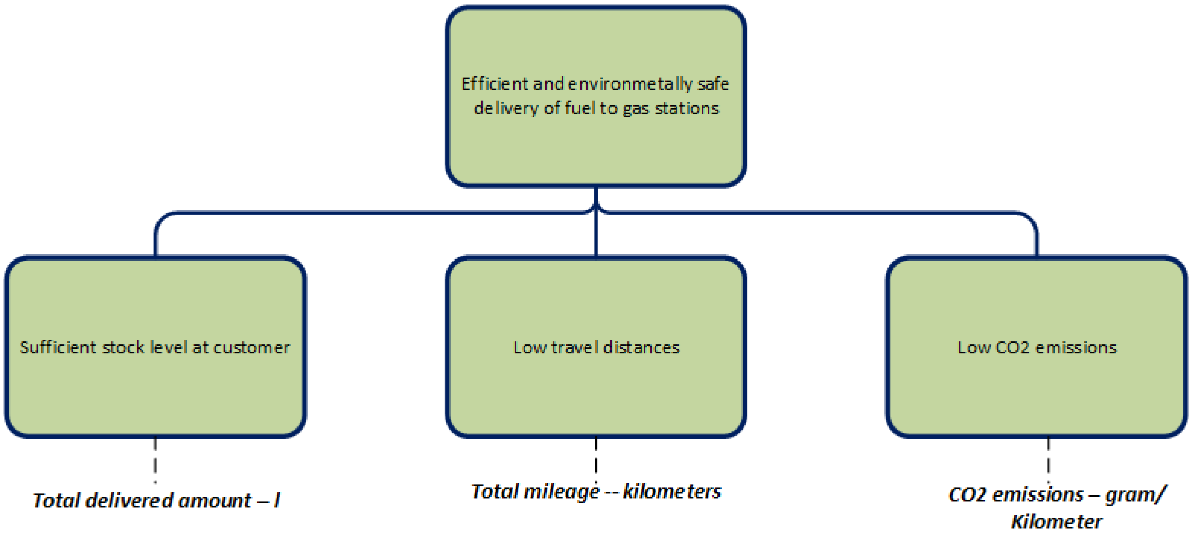

Figure 4.1: Goal tree of AMCS, including units of measurement.

4.2.1. Total mileage and CO2 emissions

One of the goals is to decrease the travel time costs. This can be translated into the total mileage that is driven by all vehicles in the system. Keeping in mind that the paths between different locations are a given and cannot change because of this, it is in fact the total mileage that will change when assessing different strategies for order generation. After all, if the total amount of trips change, there is a change that this also has an impact on the kilometers driven. The best strategy would be the one that keeps this as low as possible for a given amount of orders. Another KPI directly connected to the mileage are of course the CO2 emissions, as one of the goals is to lower the carbon footprint of transport. The mileage has a direct influence on the emissions because a vehicle emits a certain amount of CO2 per kilometer.

4.2.2. Total delivered amount

The next KPI that is necessary does have a direct relation with the goal to keep the stock levels at petrol stations as low as possible. As has already been explained in the problem description, the choice for a certain stock level can greatly influence the size of orders, thus leading to differences in vehicle and compartment utilization. Filling all tanks at a station to its maximum capacity will cause less frequent visits to that station over a specific time period, but it also means that less orders can be handled during one day, such that less stations can be handled from one depot. The ideal situation would be to deliver as little as possible, while still avoiding depletion. By looking at the total delivered amount of fuel over a certain period and compare this to the total travel distances, it is possible to see how efficient a certain strategy is in terms of amount/distance.

4.3. Supporting indicators

Some performance indicators are not important for the final outcomes, but are necessary to analyze if the model is behaving as expected. This section will briefly discuss the supporting variables.

4.3.1. Total number of trips and number of trips with x stops

Since the essence of this research is to look into different order pairing strategies, it is wise to have some kind of indicator that directly measures the changes between different order pairing strategies. This is done by checking the number of trips made by every vehicle, while also counting the amount of stops that are added to these trips. In general most vehicles will make one or two stops per trip, with an outlier of three stops when the situation allows this. When having the same type of orders for different strategies, a different amount of stops can also have a big influence on the total number of trips. Eventually this will have its impact on the total mileage and it is interesting to look how this the travel distances are impacted by the different trip sizes.

4.3.2. Utilization of vehicles

Another parameter is the Utilization of vehicles. It gives an indication how the different vehicles are utilized compared to each other, as well as the total utilization. This can give information on how well a certain strategy is performing, while it also indicates if the model is behaving as expected. If one of the vehicles is used much less than others, this could mean there is something wrong in the model. The utilization of all vehicles together give information on how well a certain order strategy is performing. Lower utilization will probably also lead to a lower mileage, given that the same amount of stations have to be served for every strategy.

4.3.3. Average utilization of compartments

By measuring the levels of fuel in the compartments it can be seen how much the compartments are actually used. If for example a few compartments are always less full than the others, it could tell something about mistakes in the decision rules. When compartments are always half full, it means that the truck could possibly have visited more station on one trip. Thus the utilization can help to interpret how well a certain strategy is performing.

4.3.4. Amount of executed orders (or total stops)

to check if the proposed model is behaving as it should be, the amount of executed orders is needed for reference. As mentioned before, travel times are influenced by a few uncertain factors and will have an impact the total amount of orders that can be delivered. If the travel times change for different scenarios, it will ultimately have its impact on the amount of orders that can be delivered during one day. This makes the total amount of executed orders an measurement for reviewing the impact of different strategies and understanding the proposed solutions. Having fewer or more orders are not necessarily good or bad with respect to each other. For example, if less processed orders lead to less trips to the most remote stations, while giving them the same amount of a product, it is not bad to have fewer orders.

4.4. Overview of KPIs and supporting indicators

The first part of this chapter has answered the question: *"What KPIs are relevant for measuring the effects of different order generation strategies and how are they measured?"*.

With the consultation of AMCS it became clear that there are four specific goals: on–time delivery, Avoiding stock buildup at the petrol stations, reducing transport costs and reduce the CO2 footprint. These goals in turn have been translated into three KPIs to measure the performance of different strategies. Furthermore four supporting variables have been defined. All KPIs and supporting variables are listed in table 4.1.

Table 4.1: Overview of relevant KPIs and supporting variables with their corresponding units.

KPI	**Units**
Total mileage	kilometers
CO2 emissions	gram/kilometer
Total delivered amount	liters
Supporting Variable	**Units**
Number of trips with x stops	trips/month
Average utilization compartments	liters
Utilization of vehicles	% of time
Amount of executed orders	orders/month

4.5. Fundamentals integrated from the AMCS case

With the exploration of the AMCS case data, algorithm and goals, it is possible to make a list of elements from the AMCS model that also form the basis of the model in this research. These elements will be discussed in this section.

4.5.1. Consumption rates

For this research the stock flows of tanks at the stations are chosen to be deterministic. One week of data about the consumption rates is given. This means that the flows in one tank will always follow the same weekly trend over the whole simulation period. In this way the situation will always be the same for every tested strategy. Obviously the consumption rates can vary between tanks and stations, since there are stations that will have more customers than others on an average day, leading to higher consumption rates. In reality fluctuations over the day can happen, like peak vs. off–peak hours, but since the tank levels will only be measured at the beginning of each day, hourly changes will not be taken into account. Knowing this, the daily differences between the data points are assumed to have a linear consumption.

4.5.2. Minimum stock levels and empty levels

The second characteristic is that tanks at stations have a certain capacity ranging from zero to several thousand liters. However, the actual level defined as empty is not equal to zero, because there should be a margin to overcome the problem of depletion if a late delivery were bound to happen. This way there is always a margin when a delay in the delivery takes place, for example when a truck is facing traffic jams. While this is accounted for in the real–life, in the scope it was stated that disruptions are not taken into consideration. Still the minimums defined by the AMCS data are taken into account, because it is always necessary to stay above this limit to avoid depletion and therefore the model will always aim for a delivery before the minimums are reached.

4.5.3. Minimum, target and additional amounts

Furthermore AMCS works with minimum, target and additional amounts. Minimum and target amounts will also be used in this model. This helps to give the optimization model some room to play with the final amounts that are loaded into a vehicle when assigning a route. The minimum amount is always 80% of the target amount. Changing the lower and upper bounds of the target amounts can lead to different results when looking at the distances. Delivering less will lead to more frequent trips, but there is a higher possibility that multiple stations can be visited on one route. Additional amounts are an option in the model, but it should be noted that additional amounts are only possible when all orders loaded on a vehicle have first reached target amounts.

4.5.4. Vehicle working schedules, station opening hours and vehicle restrictions

The model will take into account working schedules and station opening hours. This means that apart from a latest possible delivery time, defined as the time that a tank reaches depletion, the time windows are also restricted by the schedules and opening hours. Furthermore vehicle restrictions can limit the type of vehicle that is allowed to visit certain stations.

4.5.5. Time period and simulation horizon

The time period is an important factor in the research of the PSRP. Several papers have emphasized on the possible impact of multi–period instances, where the delivery of orders is optimized over several days. This can have a significant impact on the costs because the model will look for the best solution over multiple days instead of just optimizing one day of deliveries. The downside of a multi–period is found in the computing times, since multiple days mean more stations, routes and vehicles to assign.

AMCS has stated that they think the considered period can have a large impact and would like too look at a multi–period simulation of the problem. More specifically they asked to look over a horizon of 3 to 4 weeks. The data contains consumption rates for a week, so it would be possible to have a multi–period spanning several days and repeat this for four weeks, while repeating the consumption rates for every week. However, because it would probably not be achievable to do so considering the time for this project, the proposed method is to optimize order deliveries per day, but looking over a time span of 3 to 4 weeks. This can still lead to significant results, since the order pairing strategy will directly influence the impact on the order amounts, travel times and stock levels.

4.6. Requirements, objectives and constraints for the model

Based on the information provided in the previous sections, this section will give the requirements, objective and constraints for the model design.

4.6.1. Requirements

The model requirements are defined such that it is able to test different order pairing strategies over an extended period of time. It must handle all orders from one single depot towards all its client stations. The model is able to read the input data handed in a certain format and give the right outputs based on this data. Some variables have to handle manual changes, such that different outcomes are possible.

4.6.2. Objectives

The goals of AMCS, reducing transport costs and CO2 emissions, can be reached by lowering the amount of kilometers driven. It seems logical that the objective of the *order pairing optimization model* is to keep the mileage as low as possible. However, because the model works with time related variables like time schedules, station opening hours and service times, it is more logical to work with travel times. AMCS provides both an travel time and distance matrix and since the travel times and distances are normally closely related, both can be used. Eventually the outcome of the model will measure the distances of the routes as well, making it possible to compare different alternatives on their total distances. therefore the objective of the order pairing optimization model will be to minimize the total travel time of petrol deliveries.

Furthermore one should imagine the situation where one tank at a station is reaching lower limits, while another tank still has a lot of its product left. This means that the latter tank could have received less product during the last delivery. The fuel still left in the station tank when a new delivery arrives, could have been used somewhere else, which is a waste of resources. In the ideal situation every tank should have similar depletion times. therefore the objective for the *order generation model* is to aim at similar depletion times for the tanks located at the same station.

4.6.3. Constraints

One of the goals of AMCS is to provide on–time delivery to every station. This factor is considered such an important constraint for AMCS that it must always be fulfilled. As told before, on–time delivery is a certainty you provide as a petroleum company towards your clients. It is usually not acceptable that deliveries are too late, since this would lead to serious problems in the fuel supply to road traffic. therefore the model should have a hard constraint to have on time deliveries at all times. Another hard constraint is that the orders always have to be delivered in a feasible time window. This means it has to take into account station opening hours and vehicle schedules, when assigning a route during a certain time on a specific vehicle. Next to that it is not allowed for a vehicle to visit a station with restrictions on that vehicle type.

4.6.4. Overview of requirements, objectives and constraints

An overview of the specifications are given in table 4.2:

Table 4.2: Overview of model requirements, objectives and constraints.

Model requirements
Test different order pairing strategies over an extended time period.
Objectives
Order generation model: Similar depletion times for the tanks located at the same station.
Order pairing optimization model: Minimize travel times of petrol delivery to stations.
Constraints
On–time delivery to every station.
Delivery only possible in feasible time windows.
Vehicle can only visit allowed stations.

4.7. Model description and assumptions

This contains a description of the basic elements and assumptions of the model, which regards the inputs from the previous sections.

4.7.1. Overall model structure

It has become clear that the model of AMCS actually can be split into two separate models. An overview of the proposed structure of the model is visualized in figure 4.2. The model structure can be defined as simulation period with a time step of one day, where one day is split into two separate algorithms. The first model is the order generation, which checks the stock levels of all tanks at the beginning of each day, compares this with the minimum stock level allowed and makes an order for a station if one of its tanks has reached minimums. Subsequently the algorithm adjusts the amounts for all tanks at a station aiming at equal depletion times. This is the less complex part of the algorithm.

The second model does the actual order pairing, where orders are assigned to a vehicle based on amounts, locations and time windows. This second part of the algorithm actually is the most complex part, where orders have to be paired and routes need to be calculated to optimality.

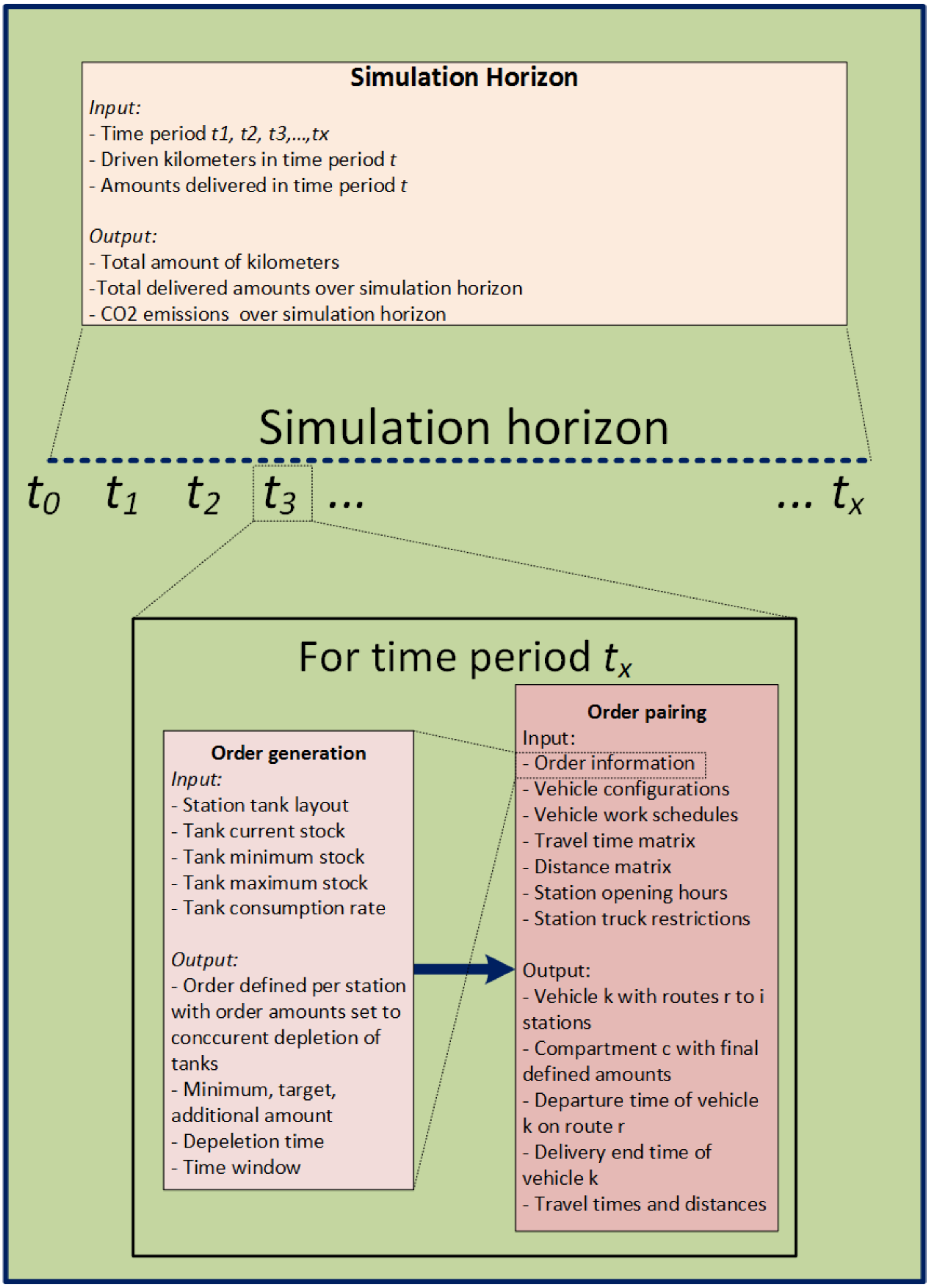

Figure 4.2: The model structure, with order generation and order pairing specified as sub models.

4.7.2. Formal description of the models and its assumptions

A network is considered where 59 stations are supplied by one depot. The Multi–Period Split Compartment Vehicle Routing Problem with Time Windows and Vehicle Restrictions (MP-SCVRPTWVR) can then be defined as follows: $G = (V, E)$ is a connected undirected graph defined as having a vertex set $V = \{0, 1, 2, ..., n\}$ with a set of edges $E = \{(i, j) \in V \times V; 0 \leq i, j \leq n, i \neq j\}$. In this case the connected graph means that each vertex is connected with all other vertices through at least one edge. Vertex zero is representing the depot, while vertices 1 through n represent the stations. All tanker trucks start and end their trip at the depot and the route can consist multiple stations.

The model works with routes, which are feasible chains of visits to a subset of stations. A route r visits all stations of the subset $V_r \subseteq V$, where the stations i are indexed according to the order in which they are visited. There are three different fuel types from two unique oil brands, which are delivered to the stations by a fleet of heterogeneous vehicles all having flow meters. The maximum inventory level for each station tank is determined as a percentage of the total volume of the tank. Furthermore all initial stock levels are known. Next to that the following assumptions are made:

- The oil depot has an infinite supply.
- A station is only allowed to be visited once per day.
- In line with the above, one order has to be delivered in total, such that it is not allowed to split an order over multiple vehicles.
- The vehicle compartment capacities cannot be exceeded.
- The inventory level of a tank can never exceed its maximum capacity. To overcome problems with depletion a safety stock level is used, which depends on the capacity and consumption of a tank.
- The stock levels of tanks are measured once at the start of each planning day and orders are generated based on this information.
- Each order has a minimum and target amount for each tank at the station corresponding to the order, which is derived from initial stock levels, expected consumption, and the maximum inventory levels of the tanks. The target amounts are derived from the maximum allowed inventory level, to push depletion as far ahead as possible, without having too much pile up at the stations. The minimum amounts are a cut down which are specified to give more possibilities for paired orders.
- The assumption above leads to an earliest delivery time. However, it is not allowed to have late deliveries that lead to stock out at the stations. Since the order should be delivered before reaching tank minimums, the depletion time is automatically the latest delivery time. This means that every station has a time window in which deliveries can take place.
- Average travel times are the same for all trucks and derived from a travel time matrix.
- Service times at the depot are 15 minutes of turnaround time and a refill time which is based on the minimum amounts that are delivered on one route, while service times at the stations are based on 10 minutes turnaround time per station plus 30 minutes unloading time for the total amount in a vehicle.
- Vehicles have regular working schedules and no overtime is allowed.
- Some vehicles may not visit certain stations. Thus some stations have vehicle restrictions.

- Since the vehicles feature flow meters, it is possible to have more than one order per compartment as long as it is the same fuel type and brand.
- With the number of compartments per vehicle it is highly unlikely that a vehicle can visit more than 3 stations. As such, one route is limited to visit up to three stations.
- During a route it is not allowed to wait at a station until the next one opens. As such only routes which can be taken in one continuous chain of travel and service times are allowed.

Just like the model of AMCS the aim for the *order generation model* is to push depletion times as far ahead as possible, without having to much stock pileup at the stations. This means it needs to forecast the expected depletion times. The *order pairing optimization model* will then use the output provided by the order generation model to find the lowest travel distances to deliver all orders. Each edge (i, j) has a specific length which is given by a distance matrix. The average travel times per link (i, j) have an important role in the routes that can be created, since the trucks have work schedules and can have multiple routes per day. The travel times are defined in a travel time matrix provided by AMCS, which is the same for every vehicle. While the travel time matrix is given in seconds, all time units are rounded to minutes. This is done since it does not make a significant difference when a route is a few seconds longer or shorter. Calculating everything in seconds would only increase calculation times, since the algorithm has to check everything by the second instead of every minute.

4.8. Overview of model assumption and implementation method

Table 4.3 gives a short overview of the assumptions that are being use in this research:

Table 4.3: Overview of model assumptions.

Model assumptions
One depot with unlimited supply
Limited heterogeneous fleet
One station visit per time period
Vehicle restrictions for certain stations
Fixed travel times and service times
Deterministic consumption rates using historical data
Station opening hours
Work schedules
Time windows based on work schedules and opening hours
No waiting times at stations allowed
Flow meters

Like stated before, the simulation model is actually a combination of two separate models. Since they use each others input and outputs they have to be able to interact with one another. This is only possible by constructing the full model using programming software. Both models are programmed using *Python*. The order pairing optimization model is implemented using *Gurobi optimization software*, which can easily be integrated with Python. As the order pairing optimization model is the complex part of the simulation model, it needs to be well structured and tested thoroughly. The next chapter will provide the mathematical model of the order pairing model and run several verification and validation tests to check if the mathematical formulation is correctly applied on paper as well as the model itself. Chapter 6 will further elaborate on the simulation model, where the order generation model is explained in more detail and the way this model interacts with the order pairing optimization model.

5

Order pairing optimization model

This chapter will give a mathematical description the order pairing optimization model. The first section will give an overview of all parameters and decision variables that are involved. Section 5.2 will further elaborate on the mathematical model defining the objective function and give the model constraints. Since the model will work with routes, a route selection procedure is applied to find feasible routes prior to starting the optimization model, which is described in section 5.3. After that section 5.4 and 5.5 will run verification and validation tests on the model. Section (5.6) will show the computational results of the model while testing several different instances with varying amount of stations, vehicles and fuel types. To make the model more efficient when handling a larger amount of orders, a modification to the current model is proposed in section 5.7. Finally, the last section will draw conclusions of this chapter.

5.1. Parameters and decision variables

Most of the mathematical model stated in this chapter uses elements from other studies done by Benantar et al. (2016), Coelho and Laporte (2015), Macedo, Alves, Valério de Carvalho, Clautiaux, and Hanafi (2011) and Li et al. (2014). Before further defining the mathematical model, the following indices and parameters are presented:

i	Index of a gas station i, belonging to set $V = \{0, 1, ..., n\}$ with the depot having index 0.
r	Index of route r, belonging to feasible routes set $R = \{0, 1, ..., r\}$ where r is a subset $V_r \subseteq V$.
k	Index of the vehicle k, belonging to set $K_1 \cup K_2 = \{1, 2, ..., k\}$.
c	Index of the compartment c, belonging to set $C_1 \cup C_2 \cup C_2^{'} = \{1, 2, ..., c\}$, where $C_2^{'} \subset C_2$.
p	Index of station fuel tank p, belonging to set $P = \{1, 2, ..., p\}$.
Q^{ck}	Capacity of compartment c of vehicle k.
L_i^{pt}	Stock level of tank p of station i at the end period t.
L_i^{p0}	Initial stock level of tank p of station i.
u_i^{pt}	Consumption rate of tank p of station i at period t.
t_i^k	The earliest time at which delivery of vehicle k at station i can take place.
$t_i^{d,k}$	The latest time at which delivery of vehicle k at station i can take place.

$v^p_{min,i}$	Minimum delivery amount for tank p of station i.
$v^p_{max,i}$	Maximum delivery amount for tank p of station i.
α_r	The earliest departure time for route r.
β_r	The latest departure time for route r.
C^p_i	Maximum capacity of tank p of station i.
$C^p_{min,i}$	Minimum capacity of tank p of station i.
$[t^a_i, t^b_i]$	The delivery time window of station i.
W^{kt}_{start}	Start time of the work schedule W of vehicle k at time period t.
W^{kt}_{end}	End time of the work schedule W of vehicle k at time period t.
H^t_i	Opening hours of station i at time period t.
S	The delivery rate, which indicates the replenishment time when a vehicle arrived at a station.
t_{ij}	Travel time from station i to station j.
M	Arbitrary large value.

To further elaborate on the variables indicated above, route r is a sequence of station visits with starting and ending at the depot. Two types of tanker trucks are defined: a semitrailer and a rigid plus drawbar combination, denoted with K_1 and K_2 respectively. Each vehicle k has a corresponding set of compartments, where K_1 belongs to compartment C_1 and K_2 belongs to compartment set C_2 or $C_2^{'}$. C_2 refers to the vehicle where the full combination rigid plus drawbar is in use. $C_2^{'}$ denotes that only the rigid compartments are used (e.g. the drawbar is detached from the vehicle), where $C_2^{'} \subset C_2$. Each compartment c has a known capacity Q^{ck}. Compartments are not dedicated to a specific fuel type such that each compartment can hold every type of fuel. Each station $i \in N$ has several tanks p from the set $P = \{1, 2, ..., p\}$, where every tank has its own fuel type. Every tank has an (initial) stock level L^{pt}_i, a consumption rate u^{pt}_i, a maximum capacity C^p_i and a minimum stock level $C^p_{min,i}$. $C^p_{min,i}$ is the minimum level that may be reached before a tank is marked as depleted. Furthermore every tank has their own minimal delivery quantity defined as $v^p_{min,i}$.

Time windows are defined as $[t^a_i, t^b_i]$, where t^b_i is the latest possible delivery time equal to the depletion time defined as the current inventory level divided by the consumption rate over consecutive periods $1, 2, ..., t$: L^{pt}_i / u^{pt}_i. The earliest delivery time is depending on the capacity of tanks and the minimum delivery allowed: $t^a_i = Max(0, (v^p_{min,i} - (C^p_i - L^{pt}_i)) / u^{pt}_i)$. The time window should always respect the opening hours of stations H^t_i and the work schedules of vehicles W^{kt}, which have a morning and evening schedule on most days.

Next to the parameters defined, the optimization model has the following decision variables:

d^k_r	The time vehicle k departs from the depot at the start of route r.
a^p_i	Amount a of fuel type p delivered to station i.
v_k	Integer value being zero if one route is assigned to vehicle k and receives +1 for every additional route on the same vehicle. If a vehicle is not used this variable becomes $-\infty$.

q_{ir}^{pck} The quantity of product p delivered to station i of route r in compartment c of truck k.

$x_r^k = \begin{cases} 1, & \text{if vehicle } k \text{ is using route } r. \\ 0, & \text{otherwise.} \end{cases}$

$R_i^k = \begin{cases} 1, & \text{if vehicle } k \text{ is allowed to visit station } i. \\ 0, & \text{otherwise.} \end{cases}$

$z_{ir}^{pck} = \begin{cases} 1, & \text{if station } i \text{ receives product } p \text{ from compartment } c \text{ of vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$

$s_{rs}^k = \begin{cases} 1, & \text{if route } s \text{ follows route } r \text{ for vehicle } k. \\ 0, & \text{otherwise.} \end{cases}$

$y_{ir}^k = \begin{cases} 1, & \text{if station } i \text{ is visited by vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$

$w_r^{pck} = \begin{cases} 1, & \text{if product } p \text{ is loaded in compartment } c \text{ of vehicle } k \text{ on route } r. \\ 0, & \text{otherwise.} \end{cases}$

$l_{r,W}^k = \begin{cases} 1, & \text{If route } r \text{ is in work schedule } W \text{ of vehicle } k. \\ 0, & \text{otherwise.} \end{cases}$

Now that the parameters and decision variables are known, the next section will further elaborate on the mathematical formulation of the order pairing optimization model.

5.2. Mathematical model

This section will provide the mathematical overview for the order pairing optimization model. While the objective of the whole model is to keep the travel distances as low as possible, the optimization model has the aim to minimize the total travel time. This is decided due to the nature of the model, where working schedules and opening times of stations are used, which makes it easier to work with (average) travel times. Because the model has the objective to minimize, this would also work through to the amounts delivered. As the model is working with minimum and target amounts, it would normally use the lowest possible delivery quantities. To overcome this problem, a penalty G is invoked for every delivered liter less than the specified target amounts. The value of G can be kept very low, as it only has the intention to penalize the deviation from its target amount. therefore the objective function and its constraints can be formulated as follows:

$$\text{Minimize} \sum_{r \in R} \sum_{k \in \{K_1 \cup K_2\}} t_r x_r^k + G \sum_{i \in N} \sum_{p \in P} (v_{max,i}^p - a_i^p) \tag{1}$$

subject to:

$$y_{ir}^k \leq R_i^k \qquad i \in N,\ r \in R,\ k \in \{K_1 \cup K_2\} \tag{2}$$

$$\sum_{k \in \{K_1 \cup K_2\}} \sum_{r \in R} y_{ir}^k = 1 \qquad i \in N \tag{3}$$

$$\sum_{i \in N} y_{ir}^{k} = \sum_{i \in r} i x_{r}^{k} \qquad k \in \{K_1 \cup K_2\},\ r \in R \tag{4}$$

$$\sum_{k \in \{K_1 \cup K_2\}} \sum_{s \in R} s_{rs}^{k} \leq 1 \qquad r \neq s,\ r \in R \tag{5}$$

$$\sum_{k \in \{K_1 \cup K_2\}} \sum_{r \in R} s_{rs}^{k} \leq 1 \qquad r \neq s,\ s \in R \tag{6}$$

$$v^{k} \geq \sum_{r \in R} x_{r}^{k} - 1 \qquad k \in \{K_1 \cup K_2\} \tag{7}$$

$$\sum_{r \in R} \sum_{s \in R} s_{rs}^{k} = v^{k} \qquad k \in \{K_1 \cup K_2\} \tag{8}$$

$$\sum_{s \in R} s_{rs}^{k} + \sum_{s \in R} s_{sr}^{k} \leq 2 x_{r}^{k} \qquad k \in \{K_1 \cup K_2\},\ r \in R,\ r \neq s \tag{9}$$

$$d_{s}^{k} + M(1 - s_{rs}^{k}) \geq d_{r}^{k} + t_{r} \qquad r \neq s,\ r \in R,\ s \in R,\ k \in \{K_1 \cup K_2\} \tag{10}$$

$$d_{r}^{k} l_{r,W}^{k} \geq W_{start}^{k} l_{r,W}^{k} \qquad k \in \{K_1 \cup K_2\},\ r \in R \tag{11}$$

$$r_{s}^{k} l_{r,W}^{k} + t_{r} \leq W_{end}^{k} l_{r,W}^{k} \qquad k \in \{K_1 \cup K_2\},\ r \in R \tag{12}$$

$$\sum_{m,e \in W} l_{r,W}^{k} = x_{r}^{k} \qquad r \in R,\ k \in \{K_1 \cup K_2\} \tag{13}$$

$$y_{ir}^{k} a_{i}^{p} \leq y_{ir}^{k} (v_{min,i}^{p} + u_{i}^{p} d_{r}^{k}) \qquad r \in R,\ k \in \{K_1 \cup K_2\},\ i \in N,\ p \in P \tag{14}$$

$$\sum_{c \in \{C_1 \cup C_2 \cup C_2'\}} q_{ir}^{pck} = a_{i}^{p} y_{ir}^{k} \qquad p \in P,\ k \in \{K_1 \cup K_2\},\ i \in N,\ r \in R \tag{15}$$

$$q_{ir}^{pck} \leq Q^{ck} z_{ir}^{pck} \qquad p \in P,\ k \in \{K_1 \cup K_2\},\ c \in \{C_1 \cup C_2 \cup C_2'\},\ r \in R,\ i \in N \tag{16}$$

$$\sum_{i \in N} q_{ir}^{pck} \leq Q^{ck} \qquad p \in P,\ k \in \{K_1 \cup K_2\},\ c \in \{C_1 \cup C_2 \cup C_2'\},\ r \in R \tag{17}$$

$$\sum_{j \in N} \sum_{p \in P} z_{jr}^{pck} = 0 \qquad i \in N,\ k \in K_2,\ c \in \{C_2 \not\subseteq C_2^{'}\},\ r \in R \tag{18}$$

$$\sum_{p \in P} w_r^{pck} \leq 1 \qquad k \in \{K_1 \cup K_2\},\ c \in \{C_1 \cup C_2 \cup C_2^{'}\},\ r \in R \tag{19}$$

$$z_{ir}^{pck} \leq y_{ir}^{k} \qquad i \in N,\ p \in P,\ c \in \{C_1 \cup C_2 \cup C_2^{'}\},\ k \in \{K_1 \cup K_2\},\ r \in R \tag{20}$$

$$z_{ir}^{pck} \leq w_r^{pck} \qquad i \in N,\ p \in P,\ c \in \{C_1 \cup C_2 \cup C_2^{'}\},\ k \in \{K_1 \cup K_2\},\ r \in R \tag{21}$$

$$x_r^k,\ z_{ir}^{pck},\ w_r^{pck},\ s_{rs}^k,\ y_i^k,\ l_{r,m,e}^k \in \{0,1\}\ i \in N,\ p \in P,\ c \in \{C_1 \cup C_2 \cup C_2^{'}\},\ k \in \{K_1 \cup K_2\},\ r \in R \tag{22}$$

In this formulation, the objective function (1) is to minimize the total travel time for visiting all stations, while the total amount delivered should be as high as possible. Constraint (2) makes sure that certain vehicles cannot visit some stations. Constraints (3) makes sure that every station is visited once and constraint (4) indicates that all stations dedicated to a route are indeed visited when this route is used. Constraint (5) and (6) limit a route combination to be used only once by any vehicle. Constraint (7) counts the amount of followup routes done by vehicle k, while (8) forces the the number of route combinations rs to be equal to the amount of followup routes in a specific vehicle. Constraint (9) limits the use of a route combination to one and only to go one way. Constraint (10) makes sure that the departure time of one route start after the arrival time of the previous route, where parameter M is an arbitrary large value. With constraints (11) and (12) it is made sure that vehicle working schedules are respected. All routes used must have a slot in one of the working schedules, meaning that the amount of slots l must equal the amount of utilized routes, specified by constraint (13).

Constraint (14) limits the maximum amount that can be delivered on a certain point in time. Starting from $t = 0$ with the minimum delivery amount, the extra amount that can be taken is proportional to the consumption over a specific time, up until the target amount is reached. The final amount that is chosen for delivery is decided on departure time with the travel time to the first station indicating the limit for the amount to be taken. With constraint (15) it is made sure that the quantities brought in all compartments is the same as the delivery amount of a specific tank. Constraint (16) only allows a quantity for a station's tank assigned to a compartment if that compartment is actually visiting the corresponding station. Constraint (17) restricts the delivery quantity in one compartment to be at most the maximum volume of that compartment. Constraint (18) sets the compartment capacity of the drawbar to zero on a specific route, if that route has a station which only allows a rigid vehicle. Constraint (19) makes sure that only one product type is assigned to one compartment. Constraints (20) and (21) allow deliveries from any compartments only if the vehicle visits a station. Constraint (22) enforces non-negative conditions on the variables.

5.3. Route selection procedure

The model described in the previous section works with predetermined routes rather than individual stations. The reason for this is the complexity of the problem, where even a small amount of stations can lead to significantly large calculation times. By constructing routes upfront it is possible to already filter out those which contain infeasible station combinations. As stated in the previous chapter, a vehicle can visit up to three stations during one route. Using permutations, all possible combinations of one, two or three stations on a route are constructed and, knowing that a vehicle always has to start and end its trip at the depot, the depot is added as a node at the beginning and end of a route. After this all routes are checked for feasible time windows between stations and if a route fits within a vehicle working schedule.

5.3.1. Arc feasibility check

The first check to lower the amount of routes is by deleting those that have infeasible station combinations, based on opening hours and delivery time windows defined in the order generation model. First the time windows of every order are adjusted to fit with the opening hours of the corresponding station. Figure 5.1 gives three examples of adjusting the time windows. The grey areas are the final time windows. The first one has a end time ($Tb1$) which is positioned within the station's opening times, but the beginning $Ta1$ of the time window starts before the station opens. The new window then has the opening time of the station as its lower bound, while the upper bound stays the original $Tb1$. The second example can fit the whole period that station i is opened into its time window. The new window will then be adjusted to have the station's opening and closure times as the new boundaries. The last example is the opposite of the previous one. Here the time window completely fits into the opening hours of the station. The time window therefore does not need any adjustments and its bounds are kept the same.

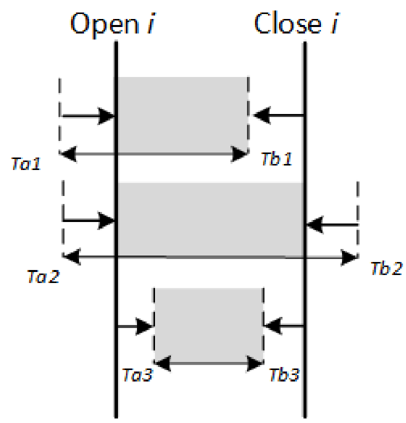

Figure 5.1: Three examples of time window adjustments with station opening hours.

After adjusting the time windows to the station opening times, the windows of different stations in a route are compared with each other for overlap. Note that it is not allowed to have waiting times at stations and the vehicle's travel time is the sum of travel times between nodes and the service times at stations. When a route is feasible, it will be provided with a departure window, which gives a lower bound α_r and an upper bound β_r for the departure of a vehicle on a certain route. This departure window takes into account the travel time to the first station to be visited for which the delivery would still be in time. As stated before in subsection 4.7.2 of chapter 4, the service times are based on an average of 60 minutes per station. For the depot there is a fixed turnaround time of 15 minutes, with an additional time based on the minimum amount a vehicle has to carry on a route divided by a filling rate of 1800 liter per minute.

Figures 5.2 to 5.5 give some examples of different situations where routes can be labelled feasible or infeasible. One situation can be that the time windows of two consecutive stations do not have overlap even while respecting the travel times between them, as shown in figure 5.2. The figure shows the first two stations in the route to share a common window while the third station is to far apart to complete this route without waiting times. This automatically leads to an infeasible route because of the waiting time restriction. As shown in figure 5.3 some of the routes can have an ordering of stations where the time window of the last station to be visited actually ends before the time window of the first station. While this specific order is infeasible, a different order could make this combination feasible which is shown in figure 5.4. The station windows are exactly the same, although this time station 3 and 1 are switched around. Because of this switch, a possible time window turns up, making this route feasible. The departure window is marked as the grey area between α_r and β_r. In figure 5.5 station 1 and three have just enough overlap to make a narrow but feasible window. The latest departure time would just be in time to reach the window's end of station 1 while the earliest departure time is exactly enough to fall within the window of station 3.

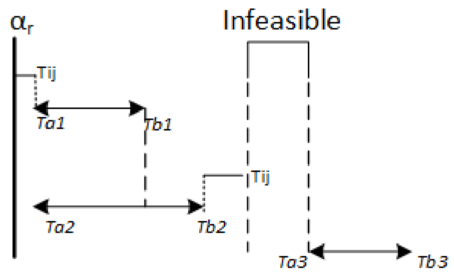

Figure 5.2: Infeasible route where station 1 and 2 have no overlap with station 3.

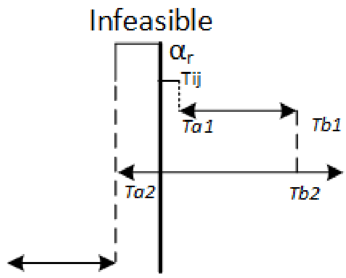

Figure 5.3: Infeasible route where the window of station 3 ends earlier than the start window of station 1.

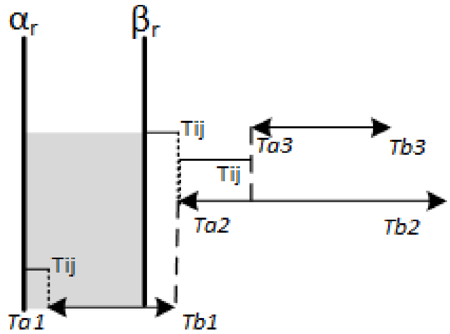

Figure 5.4: Same windows as figure 5.3 but different visiting order, making the route feasible.

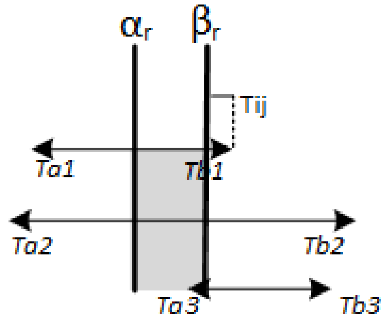

Figure 5.5: All stations have overlap making this a feasible route.

Routes with an identical subset of stations, but visited in a different order are deleted using the Pareto efficiency principle. With the knowledge that at least the minimum amounts have to be delivered to stations it becomes clear that a different order of visits will have minimal effect on the utilization of a vehicle's compartments. Additionally the amounts brought to the stations are determined on the departure time and the travel time to the first station, which would mean amounts are still equal for every station provided that both routes have the same departure time. There will only be a marginal difference in quantities if departure times are different, so overall the amounts will not change drastically when having a different order of visits. With the objective to minimize the travel times, the route with the shortest travel time will be retained, while other routes with the same subset of stations will be deleted.

5.3.2. Vehicle feasibility check

With the arc feasibility check finished, infeasible routes have been deleted and the feasible routes are assigned an earliest and latest departure time. If a vehicle is able to drive a certain route depends on two things. The first one is the restriction of certain vehicle qualifications to visit certain stations. For example when a station has little space it could be impossible for a truck with a trailer to maneuver on the terrain. The vehicle restrictions were already implemented in the optimization model itself using constraint (2). However, deleting infeasible station–vehicle combinations before starting the optimization can save a lot of time, since the model cannot assign them anyway.

An addition to the vehicle restrictions had to be made for orders which only can be delivered with a rigid vehicle. While the order generation model will always make target amounts where at least on of the tanks receives a full amount, these quantities can be too large to even fit the predetermined minimum amounts. Since the model only allows one visit per station per day some exception had to be constructed for stations with a "rigid–only" restriction. To overcome this problem, when an order can only be delivered by a rigid vehicle, the order quantity per fuel type is limited to have a target amount of 20.000 liter. If the initial amount was less than this amount, the algorithm will choose the lowest amount. Having a maximum of 20.000 liter will at most fill three tanks with the same fuel. This could still be to much when the 20% cut down is maintained, as an minimum of 16.000 liter still needs a minimum of three tanks. This would make it impossible to deliver three different fuel types if, which is very common for a station to have. therefore the maximum allowed cut down for these orders is 65%. This makes it possible to fill at most one tank with one fuel type if necessary. Basically the orders restricted to a rigid–only vehicle are given some extra breathing space to assign their orders onto the vehicle.

Furthermore it is possible to delete route–vehicle combinations based on the vehicle working schedules. If the actual departure time of a route plus the time to complete that same route do not fit in a vehicle schedule, it is an infeasible combination. An example is shown in figure 5.6, where a certain vehicle has a morning and evening schedule. The route that is compared has an earliest and latest departure time which are both starting after the morning schedule. So the route cannot be done during the morning schedule. However, for the afternoon schedule there is a possible window to fit the route into the schedule. This means that the vehicle–route combination is a feasible one.

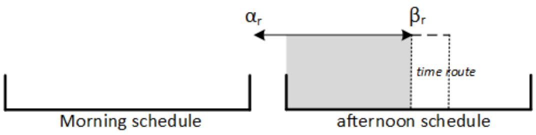

Figure 5.6: Situation where route r can be done in the afternoon schedule, but does not fit in the morning schedule. Grey area indicates departure window.

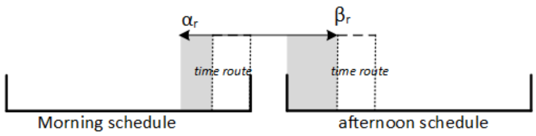

Figure 5.7: Route r fits in both the morning and afternoon schedule. Grey area indicates departure window.

The second example in figure 5.7 shows a feasible vehicle–route combination, where the departure window of the route fits in both the morning and evening schedule. Obviously if the departure window lies before the morning or behind the afternoon schedule as shown in figure 5.8 it is impossible to fit the route in the specific vehicle for that day and the this vehicle–route combination will be deleted.

Figure 5.8: Situation where the route departure window lies completely outside of a vehicle schedule, making the route–vehicle combination impossible.

5.3.3. Vehicle compartment quantity feasibility check

A last addition to the vehicle–route feasibility can be made by looking at orders which are considered to large to be paired with other orders. AMCS uses their own constraint for this, where they assume an order to be full when the target amount is at least 90% of the vehicle capacity. In this model the same assumption will be made, because the change that the remaining 10% will fit another order is significantly small, and as a result all multi–stop routes with this station as a node can be removed.

As stated before every station must at least receive their minimum order amounts. When 2 or 3 orders are combined on one route the vehicle used for the route should have enough capacity to carry the minimums of every individual order. If the capacity of a specific vehicle is not enough to do so, this vehicle–route combination is not feasible and can be removed from the set of routes before starting the optimization. The algorithm for this quantity feasibility check can be formulated as follows:

$$\text{Minimize} \quad \sum_{c\in\{C_1\cup C_2\cup C_2'\}} \sum_{p\in P} \sum_{k\in\{K_1\cup K_2\}} q^{pck} \tag{23}$$

subject to:

$$\sum_{c\in\{C_1\cup C_2\cup C_2'\}} q^{pck} w^{pck} = a^p \qquad k\in\{K_1\cup K_2\},\ p\in P \tag{24}$$

$$\sum_{p\in P} q^{pck} \leq Q^{ck} \qquad k\in\{K_1\cup K_2\},\ c\in\{C_1\cup C_2\cup C_2'\} \tag{25}$$

$$\sum_{p\in P} w^{pck} \leq 1 \qquad k\in\{K_1\cup K_2\},\ c\in\{C_1\cup C_2\cup C_2'\} \tag{26}$$

The objective function (23) will try to minimize the amounts assigned to the compartments. Basically this could also have been a maximization function, since the input is the minimum order quantity a^p which is one value that always has to be respected, which is checked with constraint (24). Constraint (25) makes sure that the compartment capacities are respected, while constraint (26) only allows one product type to be assigned on one compartment. If the compartment quantity check returns an infeasible outcome, the route–vehicle combination is discarded.

5.4. Verification of the order pairing optimization model

This section will perform some tests to see if the optimization model is showing behaviour as intended. This will help to verify if the mathematical model on paper is correct and if this is implemented correctly in the algorithm. By making predictions on how the model will behave, it can be analyzed if the implemented model has the similar outcomes and behaviour. During this process, small portions of the model will be tested separately, while the final check will engage the full model.

The model is programmed in Python using Gurobi solver and will be tested with a small data set of eight stations. The experiments are using three different objective functions: lowest travel time, highest travel time and maximizing delivery quantities. The behaviour of the sub(models) will be analyzed for these functions separately, where they will be subjected to several different experiments. The abbreviations for specific parts of the model are indicated in table 5.1.

Table 5.1: Abbreviations for models constraints and restrictions.

Model	Abbreviation
Route constraints	RC
Schedule constraints	SC
Vehicle restrictions	VR
Quantity and compartments restrictions	QC

RC is indicating that all route constraints are active, which actually is the only one always active, since it is the most basic model. Schedule constraints (SC) mean that the model must respect working schedules when assigning routes to vehicles. When vehicle restrictions (VR) are active, the model has to take into account the stations that can only be visited by a specific type of vehicle. When quantity and compartment restrictions are active, the model has to respect capacities and order quantity constraints when filling the compartments of vehicles.

5.4.1. Objective: shortest path to visit all stations

The simple version of this problem is a normal traveling salesman problem without taking into account delivery amounts. Using one vehicle and eight stations, this problem has been solved using brute–force, giving an optimal solution of 944 minutes as an comparison. The model uses weekend schedules, which means every vehicle only has a morning or afternoon schedule, but never both. The total maximum order amount for all stations is 255.687 liter. The different configurations and outcomes are seen in table 5.2.

The first two experiments should have comparable results. The first one finds the shortest route when all routes combinations are available, while the second experiment only keeps the identical routes with the shortest travel times. The third experiment adds schedule constraints while still having 1 vehicle. The expected behaviour is to get an unsolvable model because the schedule is not big enough to fit visits to all eight stations. Adding a second vehicle creates enough time to visit all stations and still creates the shortest possible route.

When adding vehicle restrictions and having two vehicles (instance 5) it is expected that at least three stations will use the rigid/drawbar vehicle. This is due to the fact that 3 out of 8 stations only allow the rigid vehicle. The result showed that four stations eventually used the rigid vehicle, where the order of one station without restrictions was combined with an order of a rigid station. The next two instances only allow one route per schedule, with the first one having 2 vehicles and the second one having 4 vehicles. Knowing that on this test day there are four working schedules in total, e.g. one for each vehicle, it is still possible to create routes that can fit all stations when having four

Table 5.2: Outcomes of models with lowest travel time objective function.

Instance	Description	Expected	Result	OK
1 RC	keep all routes, 1 vehicle	$T_{ij} = 944$	$T_{ij} = 944$	Pass
2 RC	keep shortest, 1 vehicle	$T_{ij} = 944$	$T_{ij} = 944$	Pass
3 RC+SC	keep shortest, 1 vehicle	Infeasible	Infeasible	Pass
4 RC+SC	keep shortest, 2 vehicles	$T_{ij} = 944$	$T_{ij} = 944$	Pass
5 RC+SC+VR	keep shortest, 1 rigid/drawbar, 1 semi	$\#i$ rigid ≥ 3	$\#i$ rigid $= 4$	Pass
6 RC+SC+VR	keep shortest, <3 veh., 1 route per sched.	infeasible	infeasible	Pass
7 RC+SC+VR	keep shortest, >3 veh., 1 route per sched.	feasible	feasible	Pass
8 RC+QC+SC+VR	keep shortest, 4 vehicles	$T_{ij} > 944$	$T_{ij} = 1260$	Pass
9 RC+QC+SC	keep shortest, 4 vehicles	$T_{ij} \leq 1260$	$T_{ij} = 1152$	Pass
10 RC+QC+SC+VR	keep shortest, 4 vehicles, compartment size x10, no quantity delivery time restriction	$v_{total} = 255687$	$v_{total} = 255687$	Pass
11 RC+QC+SC+VR	keep shortest, 4 vehicles, demand/10, no quantity delivery time restriction	$v_{total} = 25568$	$v_{total} = 25568$	Pass

vehicles (see instance 7). However, as expected when only having two vehicles, the model becomes unsolvable because there are only two schedules and thus two routes. Since there is a maximum of three routes per station, at most 6 stations can be fitted into this period, while 8 stations need to be visited (see instance 6).

Using vehicle restrictions also has its impact on the shortest route. In this case some route combinations become impossible. Some stations only allow rigid vehicles and because of the quantity those vehicles can take, some routes are just visiting one station to still fulfill the minimum delivery amount constraint (see instance 8). If the vehicle restrictions get removed (instance 9) but the quantity constraints are still in place, it will lead to less travel time, although it will still be higher as the shortest possible route due to the minimum amounts that have to be delivered. More capacity is needed, thus the vehicles have to make more rounds.

When the compartments are 10 times as big the carrying capacity is large enough to take the total order amounts to the stations, while still being able to take the shortest possible routes. When looking back at the experiments and its outcomes, it becomes clear that the model behaves as intended when the objective is to minimize travel time.

5.4.2. Objective: longest path to visit all stations

Following from the outcomes of the shortest route analysis it becomes clear that the model is producing the outcomes that should be expected. When aiming for the longest travel time and using one vehicle without a working schedule, the travel time will be longest when every station is visited in a single route. In this case with 10 stations the total travel time is 1319 minutes, which is around 22 hours. When choosing 2 vehicles with quantity and schedule constraints enabled, it is expected that there will be at least one route with 2 or more stations (see instance 14). This is not so much because of the delivery constraints, where it is actually easier to deliver minimum quantities when less station are visited per route. However, the schedules of both vehicles are not large enough to just choose the routes with only one station visit. Lack of time is playing its part here, and the model has to choose at least one route with 2 or more station visits. Since the objective is still aimed on the highest travel time, the outcome in this case is one route with 2 station visits. Again the model is behaving as expected. Results are shown in table 5.3.

Table 5.3: Outcomes of models with highest travel time objective function.

Instance	Description	Expected	Result	OK
12 RC	Keep shortest, 1 vehicle	$T_{ij} = 1319$	$T_{ij} = 1319$	Pass
13 RC	Keep shortest, 1 vehicle	max i per route = 1	max i per route = 1	Pass
14 RC+QC+SC	Keep shortest, 2 vehicles	routes($i \geq 2$)≥ 1	routes($i \geq 2$) = 1	Pass

5.4.3. Objective: Maximize quantities brought to stations.

For the quantity maximization objective another approach has been used to test if the model is behaving the right way. Because the quantities are depending on the compartment capacities on one hand and the tank capacities on the other hand, it is necessary to test if both these constraints are respected. Furthermore the used product can have an impact on the compartment filling. Results are shown in table 5.4.

The first experiment is the normal model with all constraints and restrictions (Instance 15). The delivered amount has to be at least the minimum amount, which is 211.758 liter. Eventually the the solution yielded 218.058 liter. When no time restrictions are applied, like with instance 16, delivery is not depending on the consumption anymore such that more quantity can be delivered, even when the delivery times would be the same. In reality this would not be realistic, since it would mean that station tanks are overflowing.

Table 5.4: Outcomes of models with maximum quantity objective function.

Instance	Description	Expected	Result	OK
15 RC+QC+SC+VR	Normal quantities	$v_{total} \geq 211758$	$v_{total} = 218058$	Pass
16 RC+QC+SC+VR	Normal quantities, no quantity time restriction	$218058 \leq v_{total} \leq 264702$	$v_{total} = 255687$	Pass
17 RC+QC+SC+VR	Huge orders 10x, no quantity time restriction	$q_{ir}^{pck} = Q^{ck}$	$Q^{ck} = 100\%$	Pass
18a RC+QC+SC+VR	Small orders /10, no quantity time restriction	$v_{total} = 25568$	$v_{total} = 25568$	Pass
18b RC+QC+SC+VR	Small orders /10, no quantity time restriction	#routes with 1 station ≤ 2	#routes with 1 station = 0	Pass
19 RC+QC+SC	Same product	Filling per comp. or equally divided until a_i^p reached	Compartments filled in order	Pass

If the orders would be increased to ten times the normal amount (instance 17), it is expected that the delivery amounts are so large that all compartments are fully filled. Indeed when conducting the experiment, all compartments were utilized for 100%. If the total amounts are divided by ten it obviously lead to almost empty compartments because it is easy to accommodate all the demanded order amounts (see instance 18a). Because the order amounts are so small, it is easier for vehicles to visit multiple stations. therefore it is expected that there are almost no routes with only one station. The model can easily find solutions where three stations are on one route and still be able to deliver the full amounts. This is confirmed by the results of instance 18b where no single station routes are used.

Interesting was to see what happens when all fuel types are the same. One of the following two things should happen; 1) the amounts are equally divided over the compartments or 2) the compartments are filled one by one until all amounts are in. None of both is wrong, but at least one should be the case when the model is implemented correctly. In this case the compartments were filled one by one. Also the vehicles were able to take larger order amounts, since the fuel types do not have to be separated. All in all the model seems to show the correct behaviour, just like with the experiments done with the other two objective functions.

5.5. Validation: comparing a test instance from literature

Since there is no current study performed which has the same characteristics as this model it is difficult to compare and validate. however, the model in this study has some identical elements with the one used by Coelho and Laporte (2015) who coincidentally provided a set of random generated instances online which they used to test their own model. One of these instances is used for validating this model. As already explained in the literature, the paper of Coelho and Laporte compares four different delivery methods to the stations, of which the split–unsplit case closely relates to this study. The differences between both models are as follows:

- in this study multiple trips per vehicle are allowed as long as they fit the vehicle's schedule. In Coelho and Laporte a vehicle is only allowed to make one route.
- This study uses vehicle restrictions and predefined routes. Coelho and Laporte uses separate stations and lets the optimization model create the routes.
- Coelho and Laporte do not use service times, while this model does.

To compare this model with the one of Coelho and Laporte the elements listed above will be filtered out to create identical circumstances. The only difference that is kept in this model will be the route generation. Creating all possible routes up front should not have any effect on the outcomes, since the only difference is that in this model routes are generated before, while the reference model of Coelho and Laporte does this during the optimization itself. Under normal circumstances travel times for the predefined routes will not differ in any way. Next to that the route pre-selection procedures will be turned off to make sure that the model can have the same route combinations as the model of Coelho and Laporte. An overview of the data used for this test instance is given in appendix A. The outcomes of both models are shown in table 5.5:

Table 5.5: Comparison of solutions between Coelho and Laporte and the model used in this study.

Coelho		
Solution	Optimization gap (percentage)	Time (seconds)
5443.75	0.00	14
Current		
Solution	Optimization gap (percentage)	Time (seconds)
5443.75	0.00	275

As can be seen the outcomes of both models are exactly the same. The only significant difference is the calculation time between them. While the model of Coelho and Laporte only needs 14 seconds to come to an optimal solution, the model used in this study needs almost 300 seconds to calculate a solution. The difference in time can be explained by the fact that Coelho and Laporte focused on certain algorithms to make the calculations quicker, which are not added in the model of this study. This is due to the different focus of both models, where Coelho and Laporte specifically looked for faster exact algorithms, while this study is aiming to look for improvements in the order pairing strategy to lower the travel distances.

Although the calculations times are significantly longer, the model does produce equal outcomes. So it can be assumed that the current model without schedules and vehicle restrictions behaves as intended.

5.6. Computational results for the order pairing optimization model

The model used in this research is unique in its combination of constraints. This makes it important to check what the added value is compared to the excising models in the literature. To explore the effectiveness of the model some tests have been conducted to see how well the model performs in terms of calculation times and solvability in general. These test were applied to the full functioning model.

Several runs will be done for one full simulation day, with a varying amount of stations, vehicles and fuel types. For example, CF–10–3–2 describes an instance with 10 stations, 3 fuel types and 2 vehicles. For every run the time limit is set to 7200 seconds. All test instances are calculated on the same computer using windows 8 operating system, with an Intel(R) Core(TM) i7–4770K 3.5GHz processor and 32GB of RAM, using two threads. The results of the different instances are shown in table 5.6. The different instances used a pre–selected set of stations with amounts based on day three from the AMCS data (which is a normal working day). More details about the stations and the order amounts of the different test instances can be found in appendix A.

The instances are chosen on feasibility, which is why the 10 station instances have 2 or 4 vehicles for example, while the 15 station instances have at least 3 vehicles. This minimum number of vehicles is the consequence of the vehicle schedules and restrictions, where too many stations for a certain type of vehicle cannot fit within the schedule of that same vehicle. All instances with 10 to 15 stations were easy to solve where all cases reached optimal solutions in minimal calculation times. The longest calculation time is around 23 seconds for the case with 15 stations, 6 fuel types and 4 vehicles. The shortest calculation times took less than a second which was the case in two of the instances with 10 stations.

The instances with 17 orders are all solvable within a reasonable amount of time, of which the one with 6 types of fuel and 3 vehicles took the longest, taking around 1.6 minutes (96 seconds) to solve. When having more than 17 stations to serve, the calculations times quickly increase and the algorithm could not find any feasible solution within 2 hours for instances with more than 19 stations. Interesting to notice is the fact that the runs with 3 instead of 6 types of fuel took longer for the model to calculate. The 20 station instance almost reached an optimum, but still showed a small gap of less than a percent, while the 21 station instance did not find any solution at all within the given time.

It was found that the combination of time schedule constraints and the range between the minimum and target amounts have a big influence on the solvability of the model, since every minute

Table 5.6: Summary of results on different test instances.

Instance	**Solution**	**Gap** (Percent)	**Time** (seconds)
CF–10–3–2	1825.5	0.00	0
CF–10–3–4	1712.5	0.00	1
CF–10–6–2	1821.8	0.00	0
CF–10–6–4	1712.5	0.00	2
CF–15–3–3	2388.3	0.00	16
CF–15–3–4	2387.7	0.00	23
CF–15–6–3	2389.5	0.00	10
CF–15–6–4	2389.0	0.00	13
CF–17–3–3	2890.5	0.00	96
CF–17–3–4	2889.9	0.00	84
CF–17–6–4	2889.9	0.00	37
CF–18–3–4	3066.5	0.00	401
CF–18–6–4	3101.5	0.00	93
CF–19–3–4	3229.1	0.00	520
CF–19–6–4	3264.5	0.00	98
CF–20–3–4	3384.6	0.21	7201
CF–20–6–4	3392.4	0.26	7201
CF–21–3–4	–	–	7201
CF–21–6–4	–	–	7202

difference in the departure time of a route also has a slight difference on the amount that can be delivered to one station. Indeed, when turning off the time schedule constraints and use a fixed value for every delivery amount the model gives feasible results for the same instances within a two hour window. In table 5.7 results are shown for the instances which were infeasible when all constraints were active. It shows that the model easily can give outcomes for the same instances in a relatively short calculation time. Note that the solutions of the 20 station instances do differ a little bit, while the routes are the same. This is due to the fact that the schedules are not a constraint anymore. It means that the model will always try to reach the latest possible departure time for every route. In other words, it will always try to get as close to the β_r of a routes departure window as possible, as long as it does not overlap with another route. This leads to less penalties for the amounts taken and thus a lower solution value.

Table 5.7: Test instances with 20 plus orders with schedule constraints turned off.

Instance	**Solution**	**Gap** (Percent)	**Time** (seconds)
CF–20–3–4	3251.6	0.00	170
CF–20–6–4	3260.4	0.00	37
CF–21–3–4	3399.0	0.00	864
CF–21–6–4	3408.1	0.00	276

5.7. Proposition: modification for the order pairing optimization model

This section will propose a modification to the order pairing optimization model. From the previous section it became clear that the scheduling constraints and ranges between target and minimum amounts caused rapidly increasing calculation times when more stations had to be served. When turning of the scheduling constraints or give one fixed amount for every order, the model was able to run much larger instances.

This observation has led to the assumption that splitting the model into two separate models could potentially lead to a more efficient model. The proposed model is actually a split model: The first one will assign routes, while the second one will fill the compartments. The first model basically is the original one with schedule constraints, but using a fixed amount for every order. Whether this are minimum or target amounts depends on the the strategy tested. So in this new model there is no range of amounts from which the model has to choose. The objective function to find the shortest travel time is kept the same, while it only can choose routes respecting the fixed amounts. The second model will then have the routes and their departure times as an input and determines how much extra fuel can be added to the minimum amounts from the previous model.

5.7.1. Mathematical changes in the split model

Since the model is split in two, there are some changes to the objective functions and constraints. In subsection 5.3.3 it was already stated that the compartment quantity check only allowed routes that could at least fit the minimum amounts. This means the route assignment model only gets routes that are already feasible with respect to the compartment capacities. therefore it is possible to drop all the compartment related constraints in the route assignment model, while all route related constraints are retained. The objective function is adapted such that the model will still look for the shortest possible travel times, but it also receives a penalty for every minute that the departure time deviates from the latest possible departure time of a specific route. The new objective function is formulated as follows:

$$\text{Minimize} \sum_{r \in R} \sum_{k \in \{K_1 \cup K_2\}} t_r x_r^k + G(\beta_r - d_r^k) \tag{30}$$

The outputs of the route assignment model are the chosen routes and its determined departure times. This is used as an input for the compartment filling model, which will determine the final amount loaded into the compartments. The minimum amounts are the same as the fixed amounts of the route assignment model and can be added up with target amounts and optional amounts, which are the delivery amounts determined when a tank would be filled to maximum capacity. The objective function for the compartment filling model is as follows:

$$\text{Maximize} \sum_{p \in P} \sum_{i \in N} a_i^p \tag{31}$$

Opposite to the route assignment model which kept its route constraints and discarded the compartment constraints, the compartment filling model will only keep its compartment related constraints. Most route constraints are discarded, except one. In the route assignment model this constraint would make sure that all station are visited once and only once. For the compartment filling model this constraint is also important, because it makes sure that the model assigns amounts for all stations on a route. The only adjustment is that the model does not have to check every route, but only the subset of routes that were chosen from the route assignment model:

$$\sum_{i \in N} y_{ir}^k = \sum_{i \in r} i \qquad k \in \{K_1 \cup K_2\},\ r \in R' \subseteq R \tag{32}$$

5.7.2. Outcomes and differences between the unsplit and split model

Some small instances of 7, 10 and 15 stations have been tested and compared for both the unsplit model and the split model. The results in table 5.8 show that the travel times of both models are exactly the same.

Table 5.8: Comparison of travel times and delivered amounts unsplit vs. split model.

Instance	**Travel time** (minutes)	**Travel time s.** (minutes)	**Diff.** (Percent)	**Amount** (liter)	**Amount s.** (liter)	**Diff.** (Percent)
CF–7–6–4	1285.9	1285.9	0.0	239561	237558	0.8
CF–10–6–4	1696.8	1696.8	0.0	292018	286485	1.9
CF–15–6–4	2207.7	2207.7	0.0	414597	399586	3.6

This is as expected since both models will always try to find the shortest set of routes, while still respecting minimum amounts. However, the total amounts delivered by both models do differ a bit. The outcomes have a difference in favor of the unsplit model ranging from less than 1 percent up till 4 percent. It seems that with increasing amount of orders the gap between the amounts slowly increase. These differences can be explained because the unsplit model is more efficient at deciding the route order to get the maximum amounts for every station, since it has information about both the schedules and amounts. The route assignment model in the split model does not use information about the amounts to decide which routes should be done first or last. It only tries to put the routes in order to get the latest possible departure times for every route, with the aim to minimize the penalty. Note that it is also beneficial for the unsplit model to let all routes depart as late as possible, since more is consumed at the stations. However, it can decide to shuffle a bit and switch some routes around to get just that little bit of extra fuel to the stations.

Overall it can be concluded that the split model is an acceptable alternative for the unsplit model. The outcomes for the travel times are the same in both models and with the knowledge that the differences in the amounts are relatively small, it is not expected to have a big impact on the simulation runs. Especially when keeping in mind that all different simulations will eventually use the same model.

5.7.3. Split model computational results

The split model has been tested for instances with up to 27 stations. The 18, 19 and 20 station instances were used to compare both models on the differences in solutions. The solution of the split model in table 5.9 represents the outcome of the route assignment model. Note that the penalties for both models are different: the unsplit model penalizes deviation from the target amounts, whereas the split model penalizes deviations between the departure times and latest departure window. This explains the slight differences between the solutions and does not have to do with the chosen routes and their travel times. For the split route assignment model the 27 station instance was the only one which could not get to optimality within 2 hours. After termination of the run there was still a gap of 0.08 percent left. Eventually it would reach convergence in at most a few hours more, meaning that the split model can come to a solution in a reasonable amount of time for an instance with at least 26 stations, compared to a maximum of 20 stations in the unsplit model. This is 25% more overall. It must be noted that the compartment filling model could always reach optimal solutions within seconds.

Looking at the calculation times there are generally large differences between both models as well. The 18 station instance has a calculation time which is already 6 times as long for the unsplit model. When taking the 20 station instance this difference already increased to 15 times as long, where the unsplit model took 2 hours to get to an non–optimized solution with a gap of 0.26%, while the split model needed only 8 minutes to come to a solution.

Table 5.9: Summary of results on different test instances.

Instance	**Solution**	**Gap** (Percent)	**Time** (seconds)	**Solution split**	**Gap split** (Percent)	**Time split** (seconds)
CF–18–6–4	3101.5	0.00	1681	2904.0	0.00	295
CF–19–6–4	3264.5	0.00	3196	3062.2	0.00	375
CF–20–6–4	3392.4	0.26	7201	3206.4	0.00	479
CF–21–6–4	–	–	7202	3245.5	0.00	583
CF–25–6–4	–	–	7201	4050.4	0.00	3874
CF–27–6–4	–	–	7201	4380.2	0.08	7200

5.8. Conclusions and discussion on the order pairing optimization model

The mathematical model used elements from other papers like Benantar et al. (2016), Coelho and Laporte (2015), Macedo et al. (2011) and Li et al. (2014). Especially Coelho and Laporte (2015) has been an valuable reference. Their paper had a reference to some test instances used in their research, of which one has also been used in this model to validate the model. However, to have identical circumstances it was necessary to turn of schedules and vehicle restrictions in this model. The test instance had exactly the same outcome as the model from Coelho and Laporte (2015). therefore it can be concluded that the model without schedules and restrictions is showing the right behaviour.

The effectiveness of the model is limited in the amount of stations that can be served. When having 20 stations or more to serve during one day, the model calculation times increase rapidly and no feasible solutions are found within a two hour time window. Making longer runs obviously could increase the changes of getting a feasible model, but it showed that the 21 stations instance was even unsolvable in a 4 day window. The bottleneck causing the rapidly increasing calculation times have been contributed to the vehicle working schedule constraints and range between amounts. When turned off the model actually got to a optimized solution for the 20 and 21 station instances in a fairly short calculation time.

An alternative approach for the order pairing optimization model was proposed, to let it handle more orders. Going from an unsplit model towards a split model, the new model first assign routes to the vehicles. If routes are assigned the compartment filling model will define the amounts that have to be loaded onto the vehicles. The split model showed to be a very promising alternative for the unsplit model, as the calculation times were greatly reduced. Next to that it is possible to get optimized solutions for instances with a higher number of stations. The split model route assignment model reached optimal solutions in two hours for runs with 25 stations. In the unsplit model this was not possible when more than 19 stations had to receive a delivery. It must be noted that the compartment filling model could always reach optimal solutions within seconds.

The next chapter will elaborate on the simulation model, explaining the functioning of the order generation model and how this is connected to the order pairing optimization model.

6
Simulation model

With the mathematical formulation, verification and validation of the order pairing optimization model being done in the previous chapter, it has to be integrated and combined with the order generation model to be able to run simulations. This chapter will first give a description about the basics, rules and assumptions of the order generation model. The next section will compare a 14 day simulation run with outcomes of a simulation provided by AMCS. In the last part a sensitivity analysis will be performed on the target amount percentage.

6.1. The order generation model

The order generation model will serve as the input for the order pairing optimization model and receives information from this same model as input. Apart from the already known variables like stations and tank configurations, it uses the consumption rates of the tanks as its main input. At the beginning of every day (00.00 hours) all the tank levels are measured and compared to the predefined minimum stock levels of those tanks. If the current stock level is equal or below the minimum stock level it will make a new order for the station corresponding to that specific tank. The order has an amount for every tank at that station, such that all tanks will be resupplied at the same time. Before further elaborating on the structure of the model the next section will briefly describe the assumptions and rules that are used in the algorithm.

6.1.1. Rules of the order generation model

To make sure that the order generation model returns orders to the optimization model that are feasible for that day, it is necessary to have some additional checks and assumptions. When the stock levels are measured at the start of each day, the model will check if one of the stock levels have reached minimum stock. The problem with this is that it can occur that a minimum stock level is reached only a few hours into the next day. This sometimes makes it impossible to assign an order to a vehicle in time. Since out–of–stock is not allowed the first check that the model performs is to look one day ahead in time. It will forecast the tank levels for the next day and check if a tank will reach minimum or empty level during that time. Then it will be determined if the minimums are reached within 32 hours. Depending on the vehicle schedule this is 2 to 3 hours into the morning schedule, which gives enough space to still plan the order on that day.

If reaching minimums takes more than 32 hours two things can happen: The order is deleted and the corresponding tank will be checked again on the next measurement point or the order is accepted based on station opening hour limitations. The latter situation occurs when an order has a delivery window plus additional travel time which ends before a station is open for delivery. next to that the end of the delivery window should not be to close to the opening times as this might lead to a time window which is to short to fulfill the order due to service times. Derived from the travel time matrix, the longest travel time is around 4 hours, and adding a safe margin of one hour, the model checks if $\beta_r < (H_{i,start} + 5)$ is true. In this case the order is shifted to the current day, as it is impossible to shift the order one day ahead due to the out–of–stock restriction.

The minimum amount for an order is a cut-down of the target amounts, which is specified to be 80% of the target amount. Next to that the minimum amount must be large enough to actually be viable to deliver. This minimum delivery requirement is the actual lower bound of an order and can differ per tank. If it turns out that the target amount was already less than the minimum specified, the minimum amount will automatically be set to zero as well. If the cut-down percentage leads to an minimum amount that is lower than 1000, but the target amount is higher than this, the minimum will be set to 1000 liters. Based on these two things a minimum amount and the earliest delivery t_i^a of the time window is generated.

6.1.2. Order generation algorithm

Figure 6.1 gives the flowchart of the order generation algorithm.

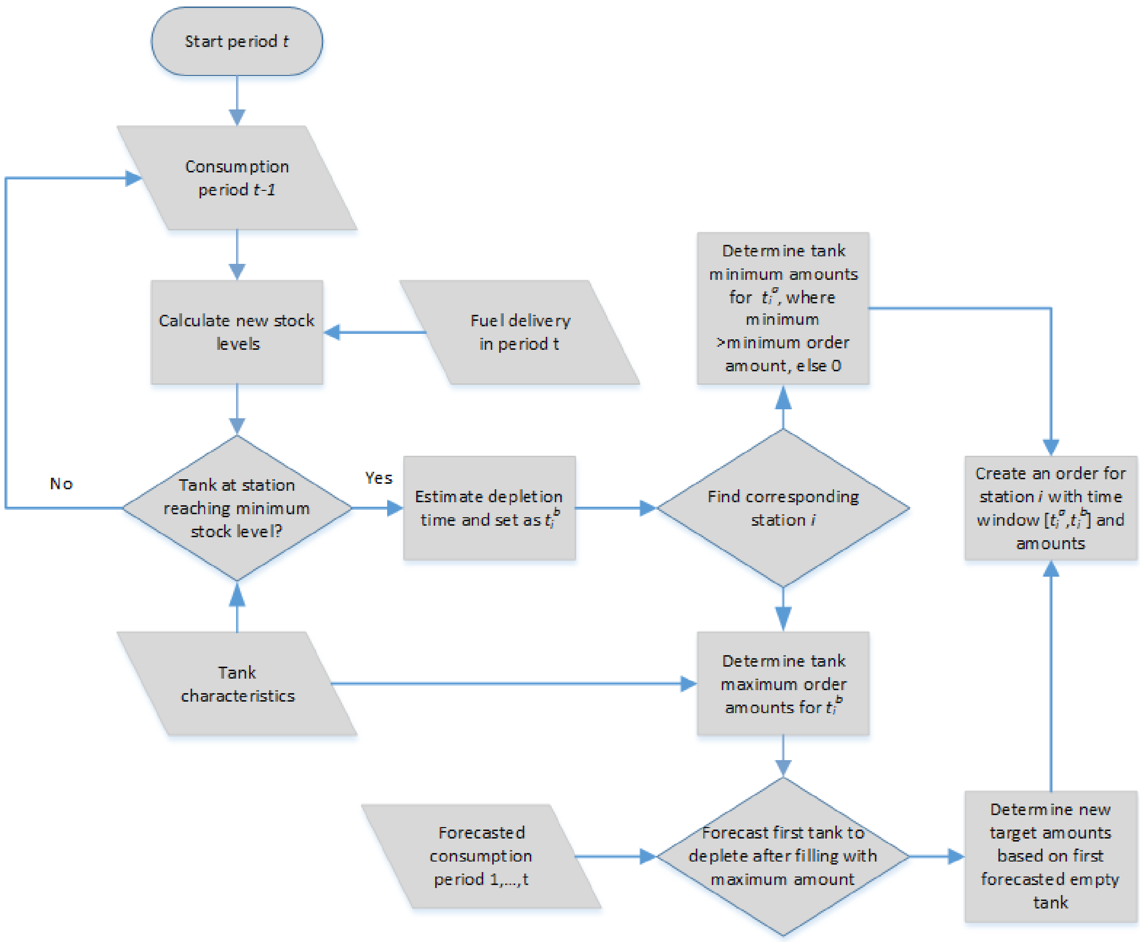

Figure 6.1: algorithm flow chart of the order generation model.

Following from the flow chart, the model starts with calculating the new stock levels. If one of the tanks at a station reaches minimum stock level, it will estimate the depletion time of that tank, which at the same time will be the latest delivery time t_i^b. Then the corresponding station of that tank is determined, where all the other tanks of that same station are receiving a target amount which is based on the maximum capacity of that tank, minus the current stock. By forecasting the consumption up until the point when one of the tanks depletes again, target amounts are adjusted to that day, aiming for similar depletion times for the next period when an order has to be made. If one of the tanks has an target amount less than the minimum delivery requirement, the order amount for that tank will be set to zero. This means that the specific tank will be eligible for a new delivery during the next time the station is visited. The following functions are necessary for defining new orders:

$$L_i^{pt} = L_i^{p,t-1} + \sum_{c \in \{C_1 \cup C_{2,3}\}} \sum_{k \in \{K_1 \cup K_{2,3}\}} q_i^{pckt} - u_i^{pt} \qquad i \in N,\ p \in P,\ t \in T \tag{27}$$

$$C_{min,i}^{p} \leq L_i^{pt} \leq C_i^p \qquad i \in N,\ p \in P,\ t \in T \tag{28}$$

$$\sum_{c \in \{C_1 \cup C_{2,3}\}} \sum_{k \in \{K_1 \cup K_{2,3}\}} q_i^{pckt} \leq C_i^p - L_i^{p,t-1} \qquad i \in N,\ p \in P,\ t \in T \tag{29}$$

Function (27) defines the inventory level of a specific fuel type at station i, where the level of a tank is determined by the tank level in the previous period minus the consumption during the time period. If a delivery takes place, this will be added to the current tank level. Constraint (28) makes sure that the inventory level is never negative or bigger than the actual capacity of a tank. Constraint (29) restricts the amount that can be delivered to a station, which can never be greater than the capacity of a tank. The latter two constraints are important for determining the amounts in an order. It makes sure that the maximum stock level can never be bigger than the tank capacity. The algorithm as implemented in the model is shown in table 6.1.

Table 6.1: Pseudo code of order generation algorithm as implemented in the program.

Algorithm Pseudo code of the order generation algorithm.

```
Decide day of week
For all tanks: subtract consumption of current day from current stock level
  If tank reaches minimum stock level or depletion level in during current day or next day. then
    Make order for corresponding station
  Else
    do nothing.
If other tank at same station is reaching minimums earlier, then
  Set new latest delivery time for station.
Else
  Keep current end delivery time.
If reaching minimums takes more than 32 hours and window day ahead does not fit station
opening hours, then
    Shift order to current day.
Else
    remove order.
For all orders.
  Add maximum and minimum amounts and earliest delivery time.
  Forecast which tank will reach minimums first.
  Adjust target amounts of other tanks to this day.
Start order pairing optimization model.
```

If all criteria are met, an order is created for station i, with a time window $[t_i^a, t_i^b]$ and the minimum and target amount bound to the earliest and latest time of the time window. This information will serve as the input of the order pairing optimization model. To prepare the optimization model for the simulation, an alternative model structure is suggested. This will be elaborated in the upcoming section.

6.2. Validation: comparison to simulation runs of AMCS

AMCS has ran its own experiments with the data they provided. Comparing the outcomes of this model with their results will help to validate if the model has comparable behaviour as theirs. The database of AMCS used data containing simulated orders for two weeks, from the 1st till the 15th of November, of which the first was a start up day. The AMCS simulation run has used similar characteristics as the model in this research. They also used the one visit per station per day constraint, as well as the restriction with the rigid/drawbar combination. Which means that just like the model used in this thesis the drawbar cannot be dropped off and picked up again anywhere during a route. The outcomes of the AMCS simulation run are shown in table 6.2.

Before showing the results it is necessary to know that there are some deviations in the data, because two stations have been removed from the data set. During the first simulation runs of the model in this research these stations, with number 10368 and 10459, caused constant errors both during the optimization runs as well as in the order generation model. They have been deleted, since it was not really known why they gave errors. Only for station 10368 can be said that it probably has to do with the rather high consumption at this station paired with the fact that the station can only receive the small rigid vehicle. This might cause problems when the vehicle cannot deliver enough fuel to keep up with the consumption. It should be possible, but in this case it might have something to do with the underlying model in this research, which cannot be fixed at the moment due to time constraints.

Table 6.2: Outcome of the 14 day simulation ran by AMCS.

Outcomes AMCS simulation (14 days).	
Driving distance	12,398
Trips	159
Stops	276
Delivered volume L	7,680,394
Volume per trip L	48,304
Avg. capacity utilization	97.8%
Stops/Trip	1.74
Dry runs	2

Trying to change their behaviour, by altering the empty levels or maximum capacity of the tanks did not change the faulty behaviour of these stations. Therefore it has been decided to leave them out of the model.

Even with the two stations removed a comparison with the AMCS run has been made. The reference run used a target amount set to 100%, which means that it will make orders aiming at fully loaded tanks at the stations. With the algorithm in this model it means that the tank which is expected to deplete first (by using forecasting) will receive a target amount that would fully fill the tank.The other tanks at the same station will have their target amounts adjusted to the same day. It should be kept in mind that without the two stations between 300.000 and 400.000 less fuel will be delivered in total (calculated from the consumption over this period). Next to that the amount of kilometers driven will be lower. Considering that on average a station will be refilled 2 to 3 times per 14 days the distances will differ between 400 and 700 kilometers.

The outcomes, given in table 6.3, do show some differences. First of all, with 11,136 km was driven, the simulation run did have a better outcome compared to the AMCS simulation with a difference of 11 percent. Even if the two stations were kept in the model, it could still be lower. This can be explained by the amount of trips that have been completed. The AMCS model did a few more trips and also had a lot more stops per trip. This means that the AMCS run had more trips of with bigger distances on average.

Interesting to see is that the average amount per trip is much higher in the AMCS model. The model run from this research had almost 8500 liter less per trip on average. The fact that some stations can only receive rigid vehicles, which can carry a maximum of 26,000 liter, does definitely contribute to this. Therefore it is surprising that AMCS was able to get such a high utilization, while applying the same rules considering rigid vehicles.

Furthermore the higher volume per trip also results in a total delivered volume which is 20 percent higher. So the amounts delivered are way lower in the model of this research. This is probably partly contributed by the two stations with high consumption which have been deleted. Another reason for the differences can be contributed to the choices for minimum and target amounts. This might differ a lot, since it is not exactly known how AMCS uses this in its algorithm. Changing these amounts can have different effects since smaller amounts could lead to more station order pairings and might even lead to higher volumes per trip because the amounts can add up easier. This could partly explain the high volume per trip of the AMCS simulation.

As already stated, the AMCS model also visits significantly more stations in general, which leads to

Table 6.3: Outcome of the 14 day simulation in this research(% change between brackets).

Outcomes simulation in this research (14 days).	
Driving distance	11,136 (–11%)
Trips	150 (–5.6%)
Stops	187 (–32.2%)
Delivered volume L	6,013,515 (–21.7%)
Volume per trip L	40,090 (–17%)
Avg. capacity utilization	88%
Stops/Trip	1.25 (–28.2%)
Dry runs	0

the suspicion that their minimum and target amounts might have different boundaries before an order is created for a station. A lot of other factors are depending on the target amount. It would be interesting to see how the model outcomes get impacted when this variable is changed. The next section will perform a sensitivity analysis on the target amount to see what the impact of this variable is on the final model run.

6.3. Sensitivity analysis on the target amount

It is assumed that the target amount can have a large influence on the model outcomes. If the target amounts are aiming for tanks that are fully filled for example, it can mean that more trips with only one order are created, since higher target amounts means bigger orders, such that pairing becomes impossible. On the other hand, having lower target amounts aiming at 70% filling of the tanks for example, will cause smaller orders where more options become available to have paired orders.

This section will test the sensitivity of this variable by using different percentages for the target amounts. The used percentage are 100, 85 and 70 percent. A 60% run has been tried, but the target amounts for some stations were becoming too low to be able to keep up with demand. After a few simulation days this would lead to an infeasible model.

The outcomes of the simulation runs are shown in tables 6.4. The first run in the table has exactly the same outcomes as the run used to compare the AMCS run and is used as a reference model.

Table 6.4: Outcomes for simulation runs with target amount set to 100, 85 and 70 percent.

	Target amount 100%	**Target amount 85%**	**Target amount 70%**
Driving distance	11,136	12,429	12,647
Trips	150	145	146
Stops	187	213	228
Delivered volume L	6,013,515	5,837,961	5,821,484
Volume per trip L	40,090	40,261	39,873
Avg. capacity utilization	88%	84%	79%
Stops/Trip	1.25	1.47	1.57
Avg. station tank utilization	61%	55%	53%

When comparing the other runs at 70 and 85 percent with the reference model, it quickly shows that the driving distances increase quite a lot. The 85% run already shows a difference of 1300 kilometer, while the differences with the 70% model is 200 kilometer more than that. It is interesting to see that the amount of trips in all models is around the same number. The biggest difference, between the 85% model and the reference model, is only 5 trips. Thereby the average stops per trip are much higher, which shows that more stops actually does not particularly lead to lower travel distances or even less trips overall.

The volumes delivered are not changing very much, although there is something interesting to notice. Lower target amounts cause less fuel to be delivered, since it is a cut down on the capacity of a tank. So obviously 100% target amount is giving higher order amounts than 85%. However the differences between 100 and 85 percent on one hand and 85 and 70 percent on the other, it shows that the decreases in delivered volumes are much lower for the latter one. The reason for this is that the model always avoids depletion. Clearly this is what is happening here, the model should always deliver a certain minimum amount to avoid depletion. This probably lies somewhere close to 5.8 million liter. And indeed, when testing for a 60% target amount it gave an infeasible model, because it could not deliver enough to keep up with consumption. This in turn leads do depleted tanks.

When referring back to the AMCS model the reference model might not actually give the best solution. The 85% run does show very promising results, where on average 1.5 orders are taken on one trip. This is a nice average and the driving distances come close to the one of AMCS. Furthermore a 100% target amount could lead to situations where the vehicles actually take to much fuel. After all, the consumption in reality is not linear as is assumed in this model. Because of this it is actually more realistic to have a safe margin of between 10 and 15 percent.

6.4. Conclusions and discussion on the simulation model

In this chapter the simulation model has been presented. It explained the functioning of the order generation model and did a validation test with a simulation case of AMCS.

When comparing the model to a simulation run of AMCS, not taking into account deleted stations, it turned out that there are quite some differences. Overall the AMCS model seems to perform quite well in terms of utilization and driven kilometers. This does not mean that the model in this research is performing worse. When for example looking at the delivered amounts, it turned out that this model scores better in having lower stock levels at stations. After all, this model delivered 1.6 million less fuel and still had no dry runs at any station, whereas the AMCS run had 2 dry runs. There are many factors that can cause the differences between both models. After all, it could just be a case of differences in assumptions and modelling constraints given by the user. Furthermore the differences are not significant enough to conclude that one of both models is better or worse.

From the sensitivity analysis follows that making more stops per trip does not particularly means less kilometers. While the model with 70 percent target amount had most stops per trip, it also had the largest travel distances. So lowering the target amounts causes less delivered volumes, but it also means more orders have to be created to keep up with the consumption. Otherwise it will lead to depleted tanks, which is not allowed. So on one hand lower target amounts cause more stops per trip, because it is easier to form order pairs. But that also means that more visits have to be made in general.

The results of the simulation model are different from the AMCS run, but together with the sensitivity analysis it seems that the model does not show behaviour that cannot be explained. This leads to the conclusion that the model in this research is good enough to be accepted for testing other strategies.

III

Results, conclusions and recommendations.

7

Strategies, results and analysis

The proposed simulation model defined in chapter 6 has been employed to test certain strategies for the case given by AMCS. This will give answer to sub question [e]: *What strategy improvements could lead to a performance increase compared to the current situation and what are the results?*. First an overview of the configuration for the simulation run will be given in section 7.1. This includes some adjustments in the data as well as some tweaks in the algorithms. After that several strategies will be proposed, which are divided into two separate subjects. Three of the strategies are aiming at lowering the calculation times of the model, while one strategy will be aiming at lower travel distances.

7.1. Simulation configuration

This section will provide an overview of the configurations in the models and the data. The first part will elaborate on the chosen simulation period and run time constraints, after which the second part of this section will explain the data modifications necessary to get a feasible simulation run.

7.1.1. Simulation time period and optimization run time

The previous chapter already tested some basic runs for a period of 14 days. For the simulation a 14 day run might be to short to see the full effects of different strategies. A longer period is necessary which is extended from 14 days towards 22 days, where the first day is a start up day, while the other 21 days are used for the analysis. Every simulation day going through the optimization model is restricted to a maximum running time of three hours. After that it will terminate and if the model is still infeasible, it will stop the whole simulation. To give the optimization model some more space, the solution is acceptable if a 0.8% gap has been reached after 3 hours or if 0.5% is reached after 2 hours. This means it could return a non optimized solution, although we can consider it being very close to the optimum.

7.1.2. Reference run

The reference run is a simulation run of 21 days with target amounts set to 85% capacity of a tank. The other model runs will also use this 85% target amount. Although the model can never have overflowing tanks due to the constraints defined in the model this could be different in a real–life case. therefore the target amount has been set to create a safe margin. The rules of the model are the same as the ones used during the validation of the simulation model. The model will try to pair order based on minimum amounts and add up more when possible.

During the 14 day simulation runs there were already some modifications in the data by deleting several stations which were giving constant errors. By doing some exploratory runs for 22 days, it was found that the amount of orders on some days caused problems. The model had some runs where it could not find a solution in three hours and therefore terminated the simulation. This was due to the fact that some simulation days, depending on the configuration used, had more than 28 orders. This was too much for the model to give a solution within the three hour window. In none of the cases with this order size it came even close to a feasible solution.

7.1.3. Adjustments in the data

To overcome problems with stations some have been removed from the final simulation runs. During the first simulation runs an infeasible day popped up at some point. Apart from the already deleted stations in the 14 day simulation, some additional stations have been removed because of this. By closely looking at the data provided by the model at the terminated day, it could be decided what to do with the stations causing problems.

Eventually it turned out that four stations caused problems during the simulation. Every time a simulation day with 28 or more orders popped up, at least 3 of these stations were also in the order list. By tweaking the minimum stock levels, maximum stock levels or changing the initial tank levels, 2 of the 4 stations could be kept in the model, because the order generation for these stations changed to other simulation days. Two stations had to be removed because their high consumption could cause orders being generated almost every other day. While it could also be an option to just lower their consumption, this does not really solve the problem but only creates the change that their will be too much orders during another day. Endless changes to the levels, amounts and consumption rates could eventually even lead to a simulation run where every station of the original data is present, but this consumes a lot of time, while skipping a few stations does not affect the testing of strategies. In principle this could also be done with half of the stations. Due to time constraints it has been decided to delete the two stations from the simulation run.

Furthermore it turned out that a weekend schedule for the vehicles also leads to infeasible runs. The vehicles have a shorter working schedule during the weekend and therefore can handle less orders than on a normal weekday. Sometimes the amount of orders was to big during the weekend. The model itself only looks ahead one day, which is limiting the abilities to foresee the amount of orders further in the future. If it is a Thursday for example, it might happen that there are only a few orders for that day, while the Saturday (without knowing up front) does have a large amount of orders. If this was known by the model it could shift back some orders to get a feasible routing schedule during the weekend. Unfortunately the model does not have this feature due to time constraints in the model building. therefore the decision has been made to give all vehicles the same schedule during the weekend, as they would have on a normal weekday.

7.2. Strategies and results

There are numerous different strategies that can be tried when looking for possible improvements. With the almost endless pool of possibilities there are too many options to consider them all. therefore four different strategies have been chosen for testing. The first one will be aiming for minimizing the travel distance. The other strategies are created to try reducing calculation times. This will be tested and compared on both the reference run as well as the outcome of strategy one.

7.2.1. Strategy 1: Target amount for largest order in a pairing

In the reference run all pairings are based on minimum amounts. While this increases the possibility of having feasible multi–stop routes, it does not mean that it is more efficient when looking at the delivered volumes and travel distances. The proposed strategy will only allow order pairings where the station with the largest order amounts takes the target amount as a minimum. The route is deleted if the specific order combination does not fit in a vehicle when the largest order uses target amount as a minimum. For the other orders on the route it only needs to reach the minimum order amounts.

The outcomes are shown in table 7.1. The first column gives the reference run of 21 days with target amounts set to 85%. When forcing the largest order on a route to have target amounts it leads to a decrease in driven kilometers of 2%. While this is not much, it shows that there is a potential benefit in using target amounts for some stations. However it does also increase the average stock levels at stations with 5 percent, which is unwanted because you want to have the station tank utilization as low as possible. The larger stations will have to be visited more frequently on average, especially when taking less amounts. It does not seem to be rewarding to aim for more order pairings when larger stations are involved. Indeed there are 5% less stops in the target order run while the amount of trips has increased with 2.5 percent.

Table 7.1: Comparison of reference run and a run using target amounts for the largest order.

Strategy:	**Reference run**	**Strategy 1 run**	**Change (%)**
Distance (km)	18996	18605	–2
Amount (l)	9257671	9398135	+1.5
Avg. station stock level (%)	56	59	+5
Avg. volume per trip (l)	39394	39258	–0.3
Avg. Capacity utilization (%)	86	87	+1
Avg. Vehicle utilization (%)	48.5	47.0	–1.5
Trips	235	241	+2.5
Stops	333	315	–5.4
stops/trip	1.4	1.3	–7
Sim. run time (s.)	4619	5205	+12.6

The volumes delivered are 2 percent higher in the target order run. This leads to a minor increase of the average utilization of compartment capacity. This compartment capacity utilization is determined by summing up the capacity of every compartment when it is used on a route. If it is used in two routes, it would mean the capacity times 2. By dividing this number with the actual loaded amount of a compartment on all routes, it gives the utilization. The average volume per trip are almost the same for both models, although slightly lower for strategy 1 run. This might be the result of having less space to pair orders in this model.

It is interesting to see that the calculation times for the target amount run are actually bigger than the reference run. Also note that the amount of stops is equal to the amount of orders over the simulation period. This means that using target amount creates less orders. There were 18 less orders made during the simulation. This potentially is a full day of deliveries. The upcoming three strategies are not only looking at the effects on the travel distances, but will also look at the calculation times. They will be tested on both the reference model and the one defined in this strategy.

7.2.2. Strategy 2: Fuel brand restrictions on routes

This strategy involves deleting every route where more than three fuel types have to be delivered on one route. With the knowledge that there are two different brands with three fuel types each, the idea behind this strategy is to only allow order pairing with stations receiving identical fuel brands. This will be a way to reduce routes and at the same time might increase the utilization of vehicles, since it is easier to combine orders.

The fuel type restriction has been tested on both the reference model and the model from strategy one. Compared to the reference run (table 7.2) the fuel type restriction does not have much impact on the reference run. Some parameters slightly increased while others have seen small decreases. The distance has increased less than a percent, while the amounts delivered also slightly went up with a small margin of 1.5%. It shows that some routes have been deleted which would otherwise have been chosen by the model. This had its impact on the calculation time, which decreased with 4.6 percent.

Table 7.2: Comparison of the reference run without and with fuel type restriction.

Strategy:	**Reference run**	**Fuel type restriction**	**Change (%)**
Distance (km)	18996	19122	+0.7
Amount (l)	9257671	9398564	+1.5
Avg. station stock level (%)	56	57	+1.8
Avg. volume per trip (l)	39394	39160	–0.6
Avg. Capacity utilization (%)	86	85	–1.0
Avg. Vehicle utilization (%)	48.5	49	+0.5
Trips	235	240	+2.0
Stops	333	336	+0.9
stops/trip	1.42	1.4	–1.4
Sim. run time (s.)	4619	4408	-4.6

Referring to table 7.3, when fuel restrictions are added to the strategy 1 run, the driven distance increases slightly, but still stays under the distance of the reference run. The calculation times are 20 percent less in the strategy 1 run, which is a large decrease.

Table 7.3: Comparison of the strategy 1 run without and with fuel type restriction.

Strategy:	**Strategy 1 run**	**Fuel type restriction**	**Change (%)**
Distance (km)	18605	18822	+1.1
Amount (l)	9398135	9567632	+1.8
Avg. station stock level (%)	59	59	0
Avg. volume per trip (l)	39258	38735	–1.3
Avg. Capacity utilization (%)	87	86	+1
Avg. Vehicle utilization (%)	47.0	47.0	0
Trips	241	247	+2.4
Stops	315	314	–0.3
stops/trip	1.3	1.27	–2.3
Sim. run time (s.)	5205	4176	-19.7

Interesting to see is that the time is even less than the runs of the reference model, which means that the fuel type restriction in combination with the tactic used in strategy 1 does show some big improvements in terms of calculation time and even slightly decreases the travel distances. The strategy 1 run with fuel restrictions does lead to a slight increase of trips, while there is one stop less. On average there are less stops per trip for the fuel restriction run. This means that in the run without fuel restrictions there are still some routes chosen where all 6 fuel types were combined, even while the target amount restriction greatly limits this possibility.

7.2.3. Strategy 3: Emphasize on three stations routes

The focus on three stop routes is an interesting option. This strategy deletes all routes with two stops on a specific vehicle if and only if all two stations on that route are also both in a feasible three stop route on the same vehicle. This leads to a lot less two stop routes and therefore gives a higher probability that the model will choose for routes with three stops.

Table 7.4 shows the results of the three route emphasis run compared to the reference run. The distances increase with 2.2 percent in this situation. The amount of trips decreased with one trip, while the amount of stops/orders increased with seven. Thus enforcing three stop routes does lead to more stops. However, just like already stated during the testing of strategy 1 it does not lead to less kilometers driven. Furthermore the total amount of delivered fuel did increase with almost 1 percent, which might be the result of three orders having better possibilities to fill a vehicle. No huge differences are found in the calculation times of both runs.

Table 7.4: Comparison of the reference run without and with three route emphasis.

Strategy:	**Reference run**	**Three route emphasis**	**Change (%)**
Distance (km)	18996	19414	+2.2
Amount (l)	9257671	9388720	+1.4
Avg. station stock level (%)	56	56	0
Avg. volume per trip (l)	39394	40123	+1.8
Avg. Capacity utilization (%)	86	85	–1
Trips	235	234	–0.4
Stops	333	340	+2.1
stops/trip	1.42	1.45	0
Sim. run time (s.)	4619	4748	+2.7

Table 7.5: Comparison of strategy 1 run without and with three route emphasis.

Strategy:	**Strategy 1 run**	**Three route emphasis**	**Change (%)**
Distance (km)	18605	18681	+0.4
Amount (l)	9398135	9474682	+0.8
Avg. station stock level (%)	59	59	0
Avg. volume per trip (l)	39258	39314	+0.1
Avg. Capacity utilization (%)	87	88	+1
Trips	241	241	0
Stops	315	317	+0.6
stops/trip	1.3	1.31	+0.8
Sim. run time (s.)	5205	4711	-9.5

The three route emphasis had little to no differences with the strategy 1 model as is shown in table 7.5. The only big difference are the calculation times between the two which is almost 10 percent. However this is only half of the run with the fuel restrictions. Which result is better depends on the most important aspect in the model. The fuel restriction run can save almost 10 to 15 minutes in calculation time, but has lead to a bigger increase of travel distances.

7.2.4. Strategy 4: Station vicinity rule

The station vicinity rule will delete a route if certain stations in that route combination are to far apart from each other. This value is predefined by the user and given in kilometers. This strategy is a very basic way to group certain stations based on geographical location. While this option could lead to a further reduction of routes, potentially improving calculation times, it might also lead to a solution which deviates a lot from the reference solution.

When applying restrictions on the distances between stations, there are lots of possibilities. In this case the distances have simply been chosen by summing up all distances and divide them by the sum of nodes, which is 59 stations plus the depot. The average came down to 42 kilometer. This is used as the boundary in the station vicinity rule. A more extreme case with a boundary of 21 kilometer has been added as well, to test if the outcomes are showing the right behaviour. It is expected that restrictions with smaller distances will lead to less paired routes. Note that in this case only the run of strategy one is used for comparison.

The results in table 7.6 and 7.7 show some interesting results.

Table 7.6: Comparison of strategy 1 run without and with station vicinity rule(21 kilometer restriction).

Strategy:	**Strategy 1 run**	**station vicinity rule (21 km)**	**Change (%)**
Distance (km)	18605	18864	+1.3
Amount (l)	9398135	9646445	+2.6
Avg. station stock level (%)	59	60	+1
Avg. volume per trip (l)	39258	35994	–8.3
Avg. Capacity utilization (%)	87	85	–2
Avg. Vehicle utilization (%)	47	42.0	–8.5
Trips	241	268	+10
Stops	315	310	–1.6
stops/trip	1.3	1.14	–12.3
Sim. run time (s.)	5205	3228	–36.1

Table 7.7: Comparison of strategy 1 run without and with station vicinity rule(42 kilometer restriction).

Strategy:	**Strategy 1 run**	**station vicinity rule (42 km)**	**Change (%)**
Distance (km)	18605	18749	+0.8
Amount (l)	9398135	9568717	+1.7
Avg. station stock level (%)	59	60	+1
Avg. volume per trip (l)	39258	37088	–5.5
Avg. Capacity utilization (%)	87	86	–1
Avg. Vehicle utilization (%)	47	43.0	–8.5
Trips	241	258	+6.6
Stops	315	312	–1
stops/trip	1.3	1.2	–7.7
Sim. run time (s.)	5205	3388	–34.9

One of the consequences of having restrictions on the distances is that routes with multiple orders become a lot more scarce when the restriction is made smaller. In the extreme case with a maximum of 21 kilometers the stops per trip decreased with more than 12 percent, coming close to a one stop per trip average. The total delivered amounts did slightly increase in both runs, with around 2 percent more delivered fuel. However the volume per trip did go down quite a lot. Especially with the

21 km restriction run, where it went down with more than 8 percent.

The most interesting outcomes of this strategy are the calculation times, which went down with a significant percentage. For both the 21 and 42 km restriction it took more than one third less of the time compared to the base run. This is more than half an hour, while the travel distances do not increase a lot. While the 21 kilometer run is a little extreme, the 42 kilometer restriction does show some promising results.

7.3. Discussion

This section will discuss the results and the models used. The first subsection will reflect on the outcomes and if those are according to expectations. The second subsection will discuss the optimization and simulation model.

7.3.1. Conclusions and reflection on the results

The outcomes of the simulation runs have shown varying results. Most results did not have any huge impact, as most variables only changed a few percent at most. Because the changes are so small it is difficult to make any hard conclusions about the outcomes. However there is some behaviour visible on all tested strategies, which might be worth looking further into when undertaking a more detailed study towards the strategies.

One of the first conclusions that can be drawn from the model is that it does not reward to actively seek for order pairing possibilities. The reference run used minimum amounts to see if certain orders could be paired. When comparing this to the run of strategy 1, where the largest order should at least have its target amount loaded, on the long run it led to less orders. This was also reflected in the amount of stops, which became 5 percent less. Eventually this resulted in almost 400 km less. The outcomes of strategy 3 does back up the finding that the focus should not be on making as much paired orders as possible. Emphasizing on the three orders per route, skipping the two order routes, caused an increase of 2 percent in the total distance traveled.

While strategy 1 seems to be a good alternative for lower travel distances, it does have the downside that the tank stock levels at stations are higher on average. The question is which one is more costly, having to more fuel stored at the stations, or driving less kilometers. Since this research does not work with costs, this is something for the user to decide.
While the main focus for strategy 1 was to lower the travel distances, the other strategies were aiming at decreasing the running times of the simulation run. Strategy 3 with the three route emphasis scored the worst of all strategies. It had the least decrease in calculation times with the strategy 1 run, and even saw an increase of almost 3 percent compared with the reference run. Next to that it was already stated that it did cause the largest increase in travel distances.

Strategy 2 with the fuel brand restrictions did show a decrease in calculation times, with the combined strategy 1–fuel restriction run needing almost 20 percent less time compared to the strategy 1 base run. This is very interesting since the driven kilometers only increased slightly and even in this situation were less than that of the reference run. Furthermore it took the combined strategy 1–fuel restriction run less time to finish the simulation. This makes the combined run of strategy 1 and 2 very efficient compared to the reference model.

Strategy 4 using station vicinity checks, is a very promising alternative for order pairing. While only a very basic version has been tested, the that the calculation times were around 30 minutes lower compared to the strategy 1 base run. The distances increased a bit, but this is not a lot when calculation times are becoming more important. This is for the client to choose. When someone wants to run a simulation within one hour, it might be rewarding to go for sub–optimized runs by adding some constraints as with strategy 2 or 4. After all, looking at the total distances over those 21 days, between 200 or 400 kilometer less travel distance is not that much. Thus it depends on the focus for the user which is more important.

One of the other things to look at was the effect of the different strategies on the carbon footprint. Only in strategy 1 the travel distances have decreased compared to the reference run. To be precise the total amount of kilometers went down with 2 percent, or 391 km. When using an average consumption of 26.5 liters per 100 km (Volvo, n.d.) and knowing that one liter of fuel combustion produces 2.6 kilograms of CO_2 (Volvo, n.d.), this leads to a reduction of 269 kg. Needless to say the reduction is far from getting near the 20% reduction to qualify for a green certification which was mentioned in the introduction chapter.

It must be noted that there was one particularly strange outcome. This was the calculation time for simulating strategy 1. It turned out to be quite high compared to the other simulations. It might be explained by the amount of order on a certain day. When the routes are selected up front, it leads to less routes with the target amount run. This is because there are less possibilities to pair orders, since there is less freedom to play with amounts. However, the simulation run is a loop which runs the split model 22 times (e.g. 22 simulation days). It might be the case that changes in the amount of orders caused some days to suddenly have a few more orders to process, where the optimization model took a while to come to a solution.

7.3.2. Reflection on the models

The starting point of this thesis was to come up with new strategies to improve the order pairing. While reviewing the case of AMCS and the literature it became clear that the model was twofold: An optimization model integrated into a simulation model. The specific requirements of the optimization model proposed in this thesis, with vehicle schedules and time windows, turned out to be too complex to handle more than 20 orders. Multiple recent problems in the papers also using time windows do acknowledge this difficulty like Cornillier et al. (2009),Li et al. (2014) and Benantar et al. (2016).

By splitting the model in two parts, the model could handle up to 28 orders in a reasonable amount of time, while not giving large differences in the outcomes compared to the unsplit model. While 28 stations are a decent amount for one run, it turned out that the data sample provided by AMCS was to large when used in a simulation run. The order generation model, controlling the amount of orders created on one day, has a lot of aspects to take into account and some of them need further improvement to really make the most out of the simulations.

There were some aspects where the behaviour of the order generation model caused the optimization model to return infeasible outcomes. The first one had to do with stations which have a high consumption. The order generation model calculates the target amounts based on the first tank to go empty again in the future. However, if this order was then send to the optimization model, it sometimes led to an order amount being to large to get even minimum amounts to fit in one vehicle. This obviously gave an infeasible outcome. A quick solution was made in the optimization model by cutting order amounts until it would fit the vehicle. The downside for this solution was that it sometimes caused the situation where the final allocated amounts were lower than the consumption, thus creating a dry run. But since the model was constrained not to have a dry run, the order generation model returned the same order the next day, but with a time window that should have been handled the day before. This caused an error in the optimization model, which could not place the order in any current day schedule. This is why some stations were deleted from the data.

8

Conclusions and recommendations

This final chapter will present the main findings and conclusions. The first section will answer the remaining sub questions and the main question. Furthermore the scientific contribution of this thesis will be highlighted, while the last part will provide recommendations.

8.1. Conclusions

This section will answer the remaining sub questions and the main question. The objective of this research was to design a model that can test different order pairing strategies for fuel delivery to petrol stations. One of the questions was: *"What KPIs are relevant for measuring the effects of different order pairing strategies and how are they measured?"*.

For AMCS the following goals are considered important: on–time delivery, avoiding stock buildup at the petrol stations, reducing transport distances and reduce the CO2 footprint. The first goal was added as a constraint in the model rather than a KPI. Depletion of tanks is not allowed and therefore is a hard boundary which may not be crossed. The second has been copied one on one as KPI in the model. The third goal is to decrease transport distances. This goal has been translated into an objective for the model. Because of this it is one of the most important KPIs to review the effectiveness of different strategies. The last goal, reduction of the CO2 footprint is directly depending on the travel distances.

The second sub question that still needs to be answered is: *"How should the structure and mathematical representation of the model design look like?"*

To test the strategies a simulation model will be used. The simulation model is divided into two sub models. The first one is the *order generation model*, which checks the stock levels of all tanks at the beginning of each day. It then compares this with the minimum stock level allowed and makes an order for a station if one of its tanks has reached minimums. Subsequently the algorithm adjusts the amounts for all tanks at a station aiming at equal depletion times.

The output of this model is used by the *order pairing optimization model*. This optimization model, aiming at minimizing travel times, is an extension to the petrol station replenishment problem. The literature, like Benantar et al. (2016), Coelho and Laporte (2015), Macedo et al. (2011) and Li et al. (2014) have addressed different combinations, but never the specific combination of station restrictions, time windows and a heterogeneous limited fleet with vehicle schedules.

The model has been validated using an test instance from Coelho and Laporte (2015) and comparing this with the results from their research. Schedule and vehicle restriction constraints had to be turned off to be similar to the model of Coelho and Laporte (2015). The test instance had exactly the same outcome in both models. therefore it can be concluded that the model without schedules and restrictions is showing the right behaviour.

While testing multiple different instances varying the number of vehicles, stations and product types, the model could reach optimality for up to 19 stations. However, having 20 or more stations became more challenging for the model to solve and showed that the 21 stations instance was even unsolvable after 4 days of running time. The bottleneck causing the rapidly increasing calculation times have been contributed to the vehicle working schedule constraints and range between amounts.

A proposition has been suggested in this research by going from an unsplit model towards a split model, which in fact are two separate optimization models. The first model assign routes to vehicles, still aiming at minimizing travel times. The second model will use these routes to define the amounts that have to be loaded onto the vehicles, with the aim to maximize delivery amounts. The split model showed to be a very promising alternative, as the outcomes for the travel times were equal to those of the unsplit model. Furthermore calculation times were greatly reduced. Next to that it is possible to get optimized solutions for instances with a higher number of stations, reaching optimal solutions in two hours for runs with 25 stations. In the unsplit model this was not possible when more than 19 stations had to receive a delivery.

The split model and generation model have been integrated to form a simulation model. The simulation model was validated and accepted by comparing the outcomes of a test run from the AMCS model. It turned out that there are some differences, like the delivered volume which was 20 percent lower. Also the amount of stops per trip was 30 percent lower. However, the model in this research did have a 11 percent lower travel distance. This has been contributed to different assumptions in both models and the results from the simulation run did not show any odd outcomes that might cause problems during a simulation run. It is concluded that the model in this research is good enough to be accepted and used for testing new order pairing strategies.

The last sub question, *"What strategy improvements could lead to a performance increase compared to the current situation and what are the results?"* is an extension of the main question. In this research four different strategies have been tested using the simulation model:

1 Target amount for largest order in a pairing.

2 Restriction of one fuel brand per route.

3 Emphasize on three stop routes.

4 Station vicinity rule.

This leads us to the answer of the main research question:

"How can the current order pairing strategy be improved and what impact does this have on the total kilometers driven and the carbon footprint?"

It is difficult to give one concise answer to this question. The four strategies presented were just a few of many options that could be tested. Furthermore, the results from the simulation runs did not show huge differences between alternatives.

One of the conclusions that can be drawn is that the user should not focus on making as much paired orders as possible. In all strategies this turned out to increase the travel distances, while the aim is to minimize them. In fact, strategy 1 seems to be a good alternative for lowering the total travel distances, by forcing the biggest order on a route to have target amounts delivered. This is an interesting finding and might lead to good solutions for other possible strategies using priority rules in the loading of order amounts.

Strategy 4 using station vicinity rules also showed promising results. While only a very basic version has been tested, it turned out that the calculation times greatly decreased with up to half an hour. The distances increased a bit, but only marginally. It depends on the goals of the user if this calculation time reduction is worth the small increase in travel distances.

8.2. Scientific contribution

The unsplit model, also referred as the Multi–Period Split Compartment Vehicle Routing Problem with Time Windows and Vehicle Restrictions (MP-SCVRPTWVR), can be categorized as a Petrol Station Replenishment Problem (PSRP). The literature has addressed some combinations, but never the specific combination of station restrictions, time windows, vehicle schedules and a heterogeneous limited fleet. To all knowledge the model constructed in this research is the first in its kind and has never been applied in a similar problem. This makes the model a contribution to the field and might prove useful for other research analyzing similar problems. Further research could be done to improve the current MP-SCVRPTWVR making it more efficient for larger sample sizes.

8.3. Recommendations for AMCS

Overall it can be concluded that more strategy testing is required to get better results out of the simulation model. Strategy 1 and 4 show the most promising results and form a good starting point to focus on. Allowing certain stations to be paired with each other based on distances is only a small example on what could be achieved with similar strategies. It might be rewarding to try grouping stations on other aspects. For example small versus large stations or stations that have the same consumption rates. This might produce certain recurring pairs that will always be paired together. This creates certainties in the order pairing, which in turn might lead to faster and better solutions.

The second option is to look at the way orders are allocated in the trucks. This could be in different sequences, for example starting with target orders for all station in a pair, or begin filling a vehicle until additional amounts of the first order. There are a lot of different combinations possible in this regard.

It is not advisable to AMCS to use the current model for a large case. The one provided for this research already proved to be too large to be analyzed in the simulation. However, the model can be used for smaller samples of up to 50 to 55 stations. This is still enough to test strategies, since those do not only depend on the amount of stations, but many other factors as well.

8.4. Recommendations on the model design

While doing some first exploratory simulation runs it was found that the amount of orders on some days caused problems. The model had some runs where it could not find a solution in three hours and therefore terminated the simulation. This was due to the fact that some simulation days, depending on the configuration used, had more than 28 orders. This was too much for the model to give a solution within the three hour window.

To make the model more robust, a few things can be done. The *order generation model* should be able to look farther ahead in time. This way the simulation model can predict how much orders will be generated on a given day and respond to this by shifting orders to another day. By doing so it is also possible to respond to weekend or holiday schedules of the vehicles. With forecasting it can also be predicted when a station needs a delivery that can still fit a vehicle. This means it will not only make an order when minimums are about the get reached, but it will also plan ahead by taking into regard the vehicle capacities.

Furthermore it might prove to be useful to give the *order pairing optimization model* more breathing space in allocating orders by allowing it to go under the minimum amounts. This could be done by penalizing this, such that it will at least try to fit the minimum amounts, but is not restricted to it. A similar thing should be done for the order generation model, as it might prove profitable to allow a dry run every now and then. This will stop the time window error, which generated an order for the dry run station, with a time window of the previous day.

Other options should be focusing on making the models faster, because it turned out to take a lot of time to calculate solutions for larger instances. This was one of the bottlenecks when trying different strategies. When one is defined, and depending on the circumstances, the model can take quite some time to run the whole simulation. If it is wanted that the strategy is tested under different circumstances, the adjustments and reruns can take up a lot of time. If calculation times are reduced, it is possible to test different strategies in a short amount of time.

Finally it must be noted that the split model used in the simulation is a model that provides sub–optimal solutions. It is divided in two separate models, thus not making full potential op the optimization process. It is recommended to improve the unsplit model, trying to make it usable for more than 20 stations and making the split model unnecessary.

References

Avella, P., Boccia, M., & Sforza, A. (2004). Solving a fuel delivery problem by heuristic and exact approaches. *European Journal of Operational Research*, *152*(1), 170–179. doi: 10.1016/S0377-2217(02)00676-8

Ballou, R. H. (2004). *Business Logistics/supply Chain Management: Planning, Organizing, and Controlling the Supply Chain*. Retrieved from http://books.google.com.co/books?id=sgsdQAAACAAJ

Barman, S. E., Lindroth, P., & Strömberg, A.-B. (2015). Modeling and Solving Vehicle Routing Problems with Many Available Vehicle Types. In A. Migdalas & A. Karakitsiou (Eds.), *Optimization, control, and applications in the information age: In honor of panos m. pardalos's 60th birthday* (pp. 113–138). Cham: Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-18567-5{_}6 doi: 10.1007/978-3-319-18567-5_6

Benantar, A., Ouafi, R., & Boukachour, J. (2016). A petrol station replenishment problem: new variant and formulation. *Logistics Research*, *9*(1), 1–18. Retrieved from http://dx.doi.org/10.1007/s12159-016-0133-z doi: 10.1007/s12159-016-0133-z

Boctor, F. F., Renaud, J., & Cornillier, F. (2011). Trip packing in petrol stations replenishment. *Omega*, *39*(1), 86–98. doi: 10.1016/j.omega.2010.03.003

Campbell, A. M., & Savelsbergh, M. W. P. (2004). Delivery Volume Optimization. *Transportation Science*, *38*(2), 210–223. Retrieved from http://pubsonline.informs.org/doi/abs/10.1287/trsc.1030.0042 doi: 10.1287/trsc.1030.0042

Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, *12*(4), 568–581. Retrieved from http://pubsonline.informs.org/doi/abs/10.1287/opre.12.4.568 doi: 10.1287/opre.12.4.568

Coelho, L. C., & Laporte, G. (2015). Classification, models and exact algorithms for multi-compartment delivery problems. *European Journal of Operational Research*, *242*(3), 854–864. doi: 10.1016/j.ejor.2014.10.059

Connekt. (n.d.). *Lean and green*. Retrieved November 16, 2016, from http://lean-green.nl/en-GB/

Cornillier, F., Boctor, F., & Renaud, J. (2012). Heuristics for the multi-depot petrol station replenishment problem with time windows. *European Journal of Operational Research*, *220*(2), 361–369. doi: 10.1016/j.ejor.2012.02.007

Cornillier, F., Boctor, F. F., Laporte, G., & Renaud, J. (2008a). An exact algorithm for the petrol station replenishment problem. *Journal of the Operational Research Society*, *59*(5), 607–615. Retrieved from http://dx.doi.org/10.1057/palgrave.jors.2602374 doi: 10.1057/palgrave.jors.2602374

Cornillier, F., Boctor, F. F., Laporte, G., & Renaud, J. (2008b). A heuristic for the multi-period petrol station replenishment problem. *European Journal of Operational Research*, *191*(2), 295–305. doi: 10.1016/j.ejor.2007.08.016

Cornillier, F., Laporte, G., Boctor, F. F., & Renaud, J. (2009). The petrol station replenishment problem with time windows. *Computers & Operations Research*, *36*(3), 919–935. doi: 10.1016/j.cor.2007.11.007

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management Science*, *6*(1), 80–91. Retrieved from http://www.jstor.org/stable/2627477 doi: 10.1287/mnsc.6.1.80

Dym, C., Little, P., & Orwin, E. (2014). *Engineering design: A project-based introduction, fourth edition.* Retrieved October 10, 2016, from http://eu.wiley.com/WileyCDA/WileyTitle/productCd-EHEP002508.html

Glover, F. (1989). Tabu Search—Part I. *ORSA Journal on Computing*, *1*(3), 190–206. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/21541959{%}5Cnhttp://pubsonline.informs.org/doi/abs/10.1287/ijoc.1.3.190 doi: 10.1287/ijoc.1.3.190

HMK Bilcon. (2016a). *City semitrailer.* Retrieved November 21, 2016, from http://www.hmkbilcon.com/en/products/fuel-tank/city-semitrailers

HMK Bilcon. (2016b). *Rigid tank trucks with drawbar trailer.* Retrieved November 21, 2016, from http://www.hmkbilcon.com/en/products/fuel-tank/rigid-tank-trucks-with-drawbar-trailer

Iori, M., & Martello, S. (2010). Routing problems with loading constraints. *TOP*, *18*(1), 4–27. Retrieved from http://dx.doi.org/10.1007/s11750-010-0144-x doi: 10.1007/s11750-010-0144-x

Li, K., Chen, B., Sivakumar, A. I., & Wu, Y. (2014). An inventory–routing problem with the objective of travel time minimization. *European Journal of Operational Research*, *236*(3), 936–945. doi: 10.1016/j.ejor.2013.07.034

Macedo, R., Alves, C., Valério de Carvalho, J., Clautiaux, F., & Hanafi, S. (2011). Solving the vehicle routing problem with time windows and multiple routes exactly using a pseudo-polynomial model. *European Journal of Operational Research*, *214*(3), 536–545. Retrieved from http://www.sciencedirect.com/science/article/pii/S037722171100381X doi: 10.1016/j.ejor.2011.04.037

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, *24*(11), 1097–1100. doi: 10.1016/S0305-0548(97)00031-2

Popović, D., Bjelić, N., & Radivojević, G. (2011). Simulation Approach to Analyse Deterministic IRP Solution of the Stochastic Fuel Delivery Problem. *Procedia - Social and Behavioral Sciences*, *20*, 273–282. doi: 10.1016/j.sbspro.2011.08.033

Popović, D., Vidović, M., & Radivojević, G. (2012). Variable Neighborhood Search heuristic for the Inventory Routing Problem in fuel delivery. *Expert Systems with Applications*, *39*(18), 13390–13398. doi: 10.1016/j.eswa.2012.05.064

Rogoff, K. (2016). *What's behind the drop in oil prices?* Retrieved November 14, 2016, from https://www.weforum.org/agenda/2016/03/what-s-behind-the-drop-in-oil-prices/

Shandong Liangshan Tongya Automobile Co., Ltd. . (2016). *Oil /fuel tanker semi trailer.* Retrieved October 21, 2016, from http://en.chinatongya.com/Product_23_0.aspx

Simchi-Levi, D. (1994). New worst-case results for the bin-packing problem. *Naval Research Logistics*, *41*(4), 579–585. Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-0028445097{&}partnerID=40{&}md5=97409bb7f130e30524a0f62d5210acd3

The Local France. (2016). *How the fuel crisis is harming the french economy.* Retrieved October 13, 2016, from https://www.thelocal.fr/20160525/how-the-fuel-crisis-will-affect-the-french-economy

Toth, P., & Vigo, D. (2002). *The vehicle routing problem* (Vol. 9). Retrieved from http://books.google.com/books?id=TeMgA5S74skC{&}pgis=1{%}5Cnhttp://books.google.com/books?hl=en{&}lr={&}id=lyukPqsBuMEC{&}oi=fnd{&}pg=PR1{&}dq=THE+VEHICLE+ROUTING+PROBLEM{&}ots=a5EJzyRTHm{&}sig=jWF4eoN5mZk7vrfwdrbit9ggRjw doi: 10.1137/1.9780898718515

Transvision. (2014). Transvision oil planner [Computer software manual].

Triki, C. (2013). Solution methods for the periodic petrol station replenishment problem. *Journal of Engineering Research*, *10*(2), 69–77.

Triki, C., & Al-Hinai, N. (2015). The Periodic Petrol Station Replenishment Problem: An Overview. In

H. A. Siddiqi, P. Manchanda, & R. Bhardwaj (Eds.), *Mathematical models, methods and applications* (pp. 127–136). Singapore: Springer Singapore. Retrieved from http://dx.doi.org/10.1007/978-981-287-973-8{_}8 doi: 10.1007/978-981-287-973-8_8

Varma, S., Wadhwa, S., & Deshmukh, S. (2008, jul). Evaluating petroleum supply chain performance. *Asia Pacific Journal of Marketing and Logistics*, *20*(3), 343–356. Retrieved from http://dx.doi.org/10.1108/13555850810890093 doi: 10.1108/13555850810890093

Vidović, M., Popović, D., & Ratković, B. (2014). Mixed integer and heuristics model for the inventory routing problem in fuel delivery. *International Journal of Production Economics*, *147*(PART C), 593–604. doi: 10.1016/j.ijpe.2013.04.034

Volvo. (n.d.). *Emissions from volvo's trucks.* Retrieved June 10, 2017, from http://www.volvotrucks.com/SiteCollectionDocuments/VTC/Corporate/About%20us/Environment-2012/Emis_eng_20640_10005.pdf

A

Overview of test instances used in the order optimization model.

A.1. Values of validation instance

Table A.1: Overview of the values used for the validation instance.

Station	Coords (x,y)	Current stock t. 1	Current stock t. 2	Demand t.1	Demand t.2
Terminal	(47,407)	–	–	–	–
1	(421,472)	10265	1653	8346	1225
2	(145,472)	6240	6456	4226	6725
3	(325,42)	7025	8261	5446	8883
4	(267,294)	2858	8627	2216	6344
5	(332,376)	1952	7074	2013	7860
6	(107,241)	10762	6257	8279	4635
7	(83,300)	7630	9349	6937	7540
8	(235,304)	3457	12177	3492	9440
9	(251,30)	9478	8665	9873	9523
10	(79,91)	7838	7347	8339	7986

A.2. Data input for computations of unsplit and split model.

A.2.1. Stations and travel time matrix

Table A.2: Travel time matrix (part one).

From/To	Depot	10171	10195	10208	10432	10435	10443	10444	10445	10450	10456
Depot	0	53	88	19	86	65	38	30	76	29	63
10171	53	0	60	47	64	68	28	33	68	43	18
10195	89	61	0	82	23	46	72	63	119	76	47
10208	20	47	82	0	79	59	31	23	72	23	56
10432	86	65	23	79	0	31	72	60	119	74	48
10435	65	69	46	58	31	0	54	39	99	53	52
10443	38	29	72	31	72	52	0	17	58	27	32
10444	29	34	63	22	60	39	18	0	64	17	37
10445	76	68	119	72	119	99	59	63	0	73	76
10450	29	43	77	22	75	54	27	19	72	0	52
10456	63	18	47	56	48	52	31	37	76	50	0
10459	48	37	44	41	44	41	31	22	77	36	20
10464	56	11	66	50	71	71	31	36	65	46	24
10466	51	42	93	47	94	73	33	38	33	48	50
10468	65	57	107	61	108	87	47	52	20	62	64
10486	48	6	59	43	62	64	23	28	63	38	16
10881	90	59	1	83	25	47	73	64	120	78	49
10924	54	2	61	48	65	69	28	33	69	44	18
10943	57	12	54	51	64	71	32	37	72	47	20
10951	33	39	85	29	85	64	24	29	53	39	50
10955	26	42	76	19	74	53	26	18	71	14	51
10972	24	41	79	10	76	56	26	20	65	26	52

Table A.3: Travel time matrix (part two).

From/To	10459	10464	10466	10468	10486	10881	10924	10943	10951	10955	10972
Depot	48	56	50	65	48	90	54	56	33	26	24
10171	36	11	42	57	5	59	2	11	39	43	40
10195	44	66	93	108	59	1	61	54	84	76	80
10208	41	50	46	61	43	83	48	51	29	19	10
10432	44	71	93	108	63	25	65	64	85	73	77
10435	41	73	73	88	66	47	70	71	64	53	56
10443	30	32	32	47	25	73	30	33	23	27	24
10444	22	37	37	52	30	64	35	38	29	17	21
10445	77	64	33	20	64	120	69	72	53	73	67
10450	37	46	46	61	39	79	44	47	37	13	26
10456	20	24	50	64	16	48	19	20	50	50	51
10459	0	42	50	65	33	45	37	39	42	36	38
10464	42	0	38	53	9	65	12	17	42	46	43
10466	51	37	0	22	37	94	43	46	27	48	42
10468	65	52	22	0	52	109	57	60	41	62	56
10486	33	9	37	52	0	58	7	10	34	38	35
10881	45	65	94	109	57	0	60	53	85	78	81
10924	37	12	42	57	6	60	0	11	40	44	41
10943	39	17	46	61	9	53	12	0	43	47	44
10951	42	42	26	41	35	86	40	43	0	39	26
10955	36	45	45	60	38	78	43	46	36	0	23
10972	38	43	39	54	36	80	41	44	26	23	0

A.2.2. Delivery windows

Table A.4: Station delivery time windows (year, month, day, hours ,minutes).

Station	Start time window	End time window
10171	2014, 11, 3, 0, 0	2014, 11, 3, 10, 12
10195	2014, 11, 3, 0, 0	2014, 11, 4, 5, 48
10208	2014, 11, 3, 0, 0	2014, 11, 4, 8, 31
10432	2014, 11, 3, 0, 0	2014, 11, 4, 5, 1
10435	2014, 11, 3, 0, 0	2014, 11, 3, 11, 38
10443	2014, 11, 3, 0, 0	2014, 11, 3, 15, 59
10444	2014, 11, 3, 0, 0	2014, 11, 3, 15, 42
10445	2014, 11, 3, 0, 0	2014, 11, 4, 20, 46
10450	2014, 11, 3, 0, 0	2014, 11, 3, 16, 5
10456	2014, 11, 3, 0, 0	2014, 11, 3, 23, 38
10459	2014, 11, 3, 0, 0	2014, 11, 4, 0, 18
10464	2014, 11, 3, 0, 0	2014, 11, 3, 20, 59
10466	2014, 11, 3, 0, 0	2014, 11, 3, 22, 11
10468	2014, 11, 3, 0, 0	2014, 11, 3, 21, 37
10486	2014, 11, 3, 0, 0	2014, 11, 3, 22, 42
10881	2014, 11, 3, 0, 0	2014, 11, 4, 0, 20
10924	2014, 11, 3, 0, 0	2014, 11, 4, 0, 40
10943	2014, 11, 3, 0, 0	2014, 11, 3, 17, 16
10951	2014, 11, 3, 0, 0	2014, 11, 3, 16, 24
10955	2014, 11, 3, 0, 0	2014, 11, 4, 4, 33
10972	2014, 11, 3, 0, 0	2014, 11, 3, 20, 55

A.2.3. Order amounts

Table A.5: Overview of station tanks with their fuel type and order amount.

Station	Tank #1		Tank #2		Tank #3		Tank #4		Tank #5	
	Fuel	Amount	Fuel	Amount	Fuel	Amount	Fuel	Amount	Fuel	Amount
10171	1030928	20215	1030928	14200	1030921	10749				
10195	1030921	9937	1030919	2017	1030928	26500	1030928	5460	1030928	7950
10208	1030900	17824	1030842	14764						
10432	1030928	26500	1030928	5720	1030921	3381				
10435	1030921	3595	1030928	26500	1030921	3951				
10443	1030921	9642	1030928	25339	1030928	7979	1030921	6043	1030919	2150
10444	1030919	1254	1030928	6278	1030928	2685	1030921	6062	1030928	4133
10445	1030900	8901	1030841	1450	1030842	5198				
10450	1030841	1639	1030900	14400	1030842	10050				
10456	1030928	8901	1030919	1533	1030921	5371				
10459	1030921	4434	1030928	13542	1030928	5933				
10464	1030928	14400	1030921	6974						
10466	1030919	3172	1030928	14400	1030921	6797	1030921	7258	1030928	9562
10468	1030841	2797	1030919	2287	1030900	23302	1030842	7791		
10486	1030928	26500								
10881	1030842	21200	1030900	26500	1030841	5905				
10924	1030900	26500	1030841	6145	1030842	26500				
10943	1030841	3568	1030842	16304	1030900	17501				
10951	1030842	15900	1030900	16358	1030841	2561				
10955	1030842	14722	1030841	2559	1030900	18000				
10972	1030928	7064	1030928	14218	1030921	26500	1030919	3231		