NTNU
Norwegian University of
Science and Technology

The Single Trade Routing Problem in Roll-on Roll-off Shipping

Torbjørn Vallestad
Aaron Weggersen

Industrial Economics and Technology Management
Submission date: July 2017
Supervisor: Kjetil Fagerholt, IØT
Co-supervisor: Marielle Chrisriansen, IØT
Jone Hansen, IØT
Jørgen Rakke, WWL

Norwegian University of Science and Technology
Department of Industrial Economics and Technology Management

Problem Description

The Single Trade Routing Problem for RoRo shipping set out to plan routes for several voyages over a planning horizon, as well as plan for which contracts to service and which quantities to load and unload at scheduled port visits. The purpose of this thesis is to further develop a mathematical formulation of The Single Trade Routing Problem by presenting comprehensive extensions, and to experiment with the resulting mixed integer programming model to gain insights for decision makers. The aim of the work will be to create a model which reflects the real problem. Therefore, the model will take into account variable vessel speeds, a heterogeneous fleet where vessels have different capacities and fuel usage, available stowage space estimation, fairly evenly spread requirements, transit time requirements, and differences in capacity and space utilization ability for different decks within a single vessel. According to the best of our knowledge, little previous research has been done on this problem, and some of our formulations contributes to existing literature by presenting new ways to model several issues.

Preface

This master's thesis is written as a part of our MSc. in Industrial Economics and Technology Management at the Norwegian University of Science and Technology (NTNU), Department of Industrial Economics and Technology Management. The thesis is a continuation of the work that was done in our specialization project during the fall of 2016.

We introduced the single trade routing problem in Roll-on Roll-off shipping with an accompanying simplified model for solving the exact solution in the project. In this thesis, we adapt real life aspects regarding the problem into the model to make a more valuable solution method for decision makers.

The work is done in a collaboration between Wallenius Wilhelmsen Logistics and the Norwegian University of Science and Technology.

We would like to express gratitude towards our supervisors Kjetil Fagerholt, Marielle Christiansen, and Jone R. Hansen for their enthusiasm, guidance, and sincere interest in our work. We would also like to thank our industry contact Jørgen G. Rakke for his valuable knowledge of the RoRo liner shipping industry and for his contributions to the project.

Trondheim, July 2017

Torbjørn Vallestad

Aaron Weggersen

Abstract

Roll-on/Roll-off shipping is the primary method for transporting vehicles and other types of material which can be rolled onto vessels, over long distances. Low demand and excess capacity have caused global shipping margins to fall to a historic low. However, comparatively little operational research has been done on RoRo shipping, suggesting there is room for improvement. The research done in this thesis is motivated by the conviction that a good decision support system has the potential for significant benefits to RoRo providers in maritime transportation.

In this thesis, we focus on operational decisions related to planning routes for several voyages over a single trade route, as well as plan for which contracts to service and which quantities to load and unload at scheduled port visits. Solving this problem for a single trade route will be an essential building block in solving the problem for several trade routes with dependencies between them.

The objective of this thesis is to gain insights that can help decision makers to reduce and minimize costs. Hence, the aim of the work will be to create a model which reflects the real problem. A mathematical formulation of The Single Trade Routing Problem has been further developed, by introducing new and comprehensive extensions. The model will take into account variable vessel speeds, a heterogeneous fleet where vessels have different capacities and fuel usage, available stowage space estimation, fairly evenly spread requirements, transit time requirements, and differences in capacity and space utilization for different decks within a single vessel.

Computational results show that the mixed integer programming (MIP) model is able to solve small instances within a reasonable time limit. In addition, the results show promising results for using the model as a decision support system, facilitating decision makers to reducing shipping costs.

Sammendrag

Roll-on/Roll-off shipping er den primære metoden for transport av kjøretøy og andre typer objekter som kan rulles på skipet, over lange avstander. Lav etterspørsel og overflødig kapasitet har ført til at globale fraktmarginer faller til et historisk lavt nivå. Det har imidlertid blitt gjort relativt lite forskning innen optimering av RoRo-shipping, noe som tyder på at det er rom for forbedring. Forskningen i denne oppgaven er motivert av overbevisningen om at et godt beslutningsstøttesystem har potensial for store fordeler for RoRo-leverandører i maritim transport.

I denne masteroppgaven fokuserer vi på operasjonelle beslutninger knyttet til det å planlegge flere reiser over en enkelt handelsrute, samt en plan for hvilke kontrakter som skal betjenes og hvilke mengder som skal lastes og losses ved planlagte portbesøk. Å løse dette problemet for en enkelt handelsrute vil være en viktig byggestein for å løse problemet for flere handelsruter med avhengigheter mellom hverandre.

Målet med arbeidet er å få innsikt som kan hjelpe beslutningstakere å redusere og minimere kostnadene. Derfor vil målet med arbeidet være å skape en modell som gjenspeiler det virkelige problemet. En matematisk formulering av The Single Trade Routing Problem er videreutviklet ved å introdusere nye og omfattende utvidelser. Modellen vil ta hensyn til variabel hastighet på skip, en heterogen flåte hvor fartøy har forskjellig kapasitet og drivstofforbruk, tilgjengelig estimat av størrelse på lasterom, forholdsvis jevn spredning av kontrakter, transittidskrav og forskjeller i kapasitet og lastutnyttelse for forskjellige dekk i et enkelt fartøy .

Beregningsresultater viser at MIP-modellen (Mixed Integer Programming) kan løse små instanser innen rimelig tid. I tillegg viser resultatene at det er lovende resultater for å bruke modellen som et beslutningsstøttesystem, slik at beslutningstakere kan redusere fraktkostnadene.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, relevant background information and an overview of maritime transportation are given. We also provide purpose and motivation for studying the problem presented. The first section, Section 1.1, a short background is given of the shipping industry with different shipping categories, modes of operation, and levels of planing, as well as some common shipping terminology. Section 1.2 contains motivation for researching the given problem. A short description of the problem is given in Section 1.3 while Section 1.4 gives an outline for the thesis.

1.1 The shipping industry

Maritime transportation

Maritime transportation is a combination of shipping and port facilities. There are many different modes of shipping, where each mode differs. Maritime transportation is the obvious choice when transporting heavy industrial machinery, especially when large volumes are transported over vast distances. This is because maritime transportation has the lowest per unit cost of all modes of transportation (Christiansen, Fagerholt, Nygreen, and Ronen, 2013). Ships operate between ports used to load and unload cargo. The ports are also used to restock supplies the vessels might need. Without this way of transportation, the necessary intercontinental scale of trade would not be possible. As International Chamber of Shipping (2016) states: "Without shipping, the import/export of affordable food and goods would not be possible - half the world would starve, and the other half would freeze!" In other words, maritime transportation is the backbone of globalization and enables international trade. International Chamber of Shipping (2016) also state that maritime transportation is the least environmentally damaging form of commercial transport.

Maritime transportation is the largest mean of international trade with an estimated range of 65% to 85% of total weight transported (Christiansen, Fagerholt, Nygreen, and Ronen, 2007). IMO (2016) estimates that maritime transportation now covers over 90% of the world's trade.

According to UNCTAD (2016), the world fleet grew by 3.5% in 2015. This is a higher growth than the 2.1% growth seen in demand for the same year. A higher growth in capacity than demand leads to a global overcapacity. UNCTAD (2016) shows that most shipping segments suffered historic low levels of freight rates and weak earnings in 2015. Carriers continued in 2015 to look for measures to improve efficiency and optimize operations. This is one reason why the field of maritime transportation is being studied by operations research.

Categories of the shipping market

Different vessels can carry various types of products. Everything from high volume liquid and dry bulk to cargo that is rolled or lifted on the vessel. UNCTAD (2016) categorize the different segments into the categories of Oil tanker, Bulk carriers, General cargo ships and Container ships as the four main categories. Table 1.1 shows the world fleet by the different types of vessels. Dry bulk carriers take up the biggest share of the world fleet. Another interesting observation is that the number of general cargo ships have steadily decreased at the same rate as container ships have increased.

	1980	**1990**	**2000**	**2010**	**2015**	**2016**
Other	4.5	7.5	9.4	7.2	11.1	11.3
Container ship	1.6	3.9	8.0	13.3	13.1	13.5
General cargo ship	17.0	15.6	12.7	8.5	4.2	4.2
Dry bulk carrier	27.2	35.6	34.6	35.8	43.6	43.1
Oil tanker	49.7	37.4	35.4	35.3	28.0	27.9

Table 1.1: The world fleet by principal vessel type, 1980–2016 (Percentage share of carrying capacity of a ship in tonnage). Source: UNCTAD (2016)

From Table 1.2, we can see that containers and dry cargo accounts for 70.7% of the total volume transported by sea. The amount of oil and gas traded the last decade has more or less stayed the same. What takes up most of the increase in total seaborne trade, is the increase in dry cargo and containers.

	1980	1990	2000	2010	2015
Containers	102	234	598	1280	1687
Dry cargo*	1123	1031	1928	2022	2463
Main bulk commodities	608	988	1295	2335	2951
Oil and gas	1871	1755	2163	2772	2947

Table 1.2: International seaborne trade for selected years. Values are in millions of tons loaded. Source: UNCTAD (2016). *Dry cargo other than main bulk commodities.

Modes of operation

One common way of structuring maritime transportation is to divide it into three modes of operation; Industrial, Tramp, and Liner (Lawrence, 1972). Industrial operators are usually the owners of the cargo they transport and control the vessels used (Christiansen et al., 2007). These companies perform more of the supply chain and can transport the necessary goods between ports depending on supply and demand at a given time with the objective to minimize costs. Liquid and dry bulk are most transported in industrial shipping.

In Tramp and Liner shipping, the sole purpose is to maximize the profit of transportation of goods, not production. Tramp shipping has its similarities to industrial shipping since supply and demand will dictate where ships sail and when. A Tramp shipping company has no set route or schedule and is usually available on short notice. Maritime transportation using tramp shipping is often compared to taxi service (Christiansen et al., 2007). There is also large flexibility when it comes to what ports can be handled since there are no fixed port calls.

Liner shipping, on the other hand, structure maritime transportation in a different way. Liner shipping has a given trade route with a fixed itinerary and schedule announced in advance. The comparison used for liner shipping is usually a bus line service (Christiansen et al., 2007). Liner shipping companies provide valuable predictability and structure to customers. Since trade routes are performed after a given schedule, it makes it convenient to plan after.

Trade routes and voyages are a big part of liner shipping. What follows is a brief explanation some central elements within maritime transportation and Roll-on Roll-off shipping.

Figure 1.1: Two different voyages in a trade route

Levels of planning

Making decisions is an important part of planning in maritime transportation. These decisions are usually divided into three planning levels depending on the complexity and time horizon they have. The different planning levels are Strategic (long-term), Tactical, and Operational (short-term) with planning horizons of years, months, and weeks or days respectively.

A strategic decision is a base on which tactical and operational decisions are built upon according to Christiansen et al. (2007). Strategic decisions are not easy to change, and implementing changes takes a lot of time. Examples of decisions to be made on the strategic level are optimal fleet sizing, optimal fleet composition, and route network design.

At the tactical planning level, the planning horizon is usually in months but can be as long as up to a year. Typical problems that are being solved the tactical level are fleet deployment, transshipment, and routing and scheduling (Christiansen et al., 2007).

In a dynamic setting with high uncertainty when decisions only have a short-term impact, operational planning is useful (Christiansen et al., 2007). Examples for the operational level of decisions are sailing speed and stowage.

Christiansen et al. (2007) states that it is not always easy to differentiate between the planning levels. The long-term decisions are the building blocks for a good short-term decision, while one usually needs detailed information about the short-term decisions in order to make good long-term decisions.

Roll-on roll-off vessels

Roll-on Roll-off (RoRo) vessels are part of the general cargo ships category, and they usually operate in the liner shipping mode of transportation (UNCTAD, 2015). There are different types of RoRo ships, including, but not limited to, RoRo passenger, Pure car and truck (PCTC) carrier and RoRo vessels.

We are focusing on RoRo vessels in this report. RoRo vessels have ramps for cargo with wheels to be rolled on and off the vessel. While cargo on container vessels are limited by the size of the container, RoRo vessels are capable of carrying cars, trucks and large construction and agricultural machinery to airplane wings, generators, and locomotives (Logistics, 2016b).

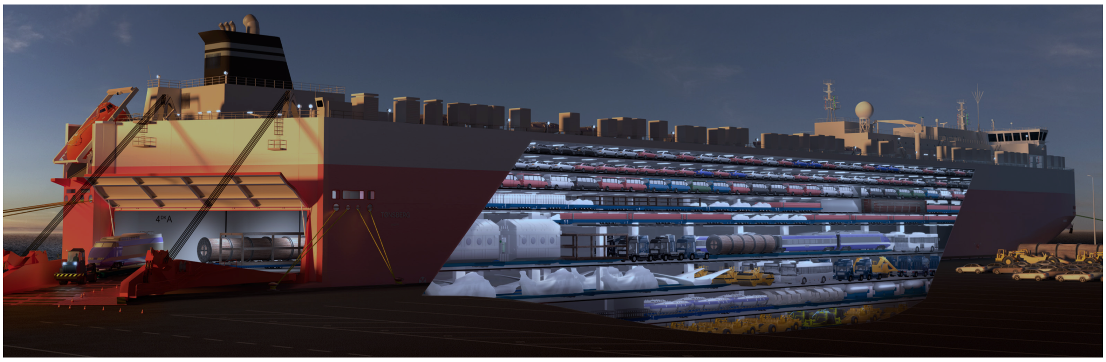


Figure 1.2: An example of a RoRo vessel. Source: Logistics (2016b)

The inside of a RoRo vessel works like a parking garage (Logistics, 2016a), as shown in Figure 1.2. There are upper decks that are designed to store cars, main decks in the middle to transport heavy, long and wide products, and lower decks to transport break bulk and rolling equipment. Vessels in RoRo shipping typically come with hoistable decks to accommodate for differences in product sizes. Hence vessel capacity, both in area and volume, can be adjusted to accommodate different needs.

Terminology

Trade route

A trade route connects two geographical regions together. In maritime transportation, the geographical regions are usually continents. Each region contains a selection of ports. Figure 1.1 illustrates an example of a trade route between North America and Europe. All the blue ports in North America and Europe are part of the trade route.

Deep sea leg

Trade routes that connects continents, typically will have what is called a deep sea leg. A deep sea leg is the section of the trade route over the ocean separating two continents. In normal mode of operation, vessels will only cross the deep sea leg once per voyage. In Figure 1.3, the deep sea leg of that particular trade route has been marked with a blue line.

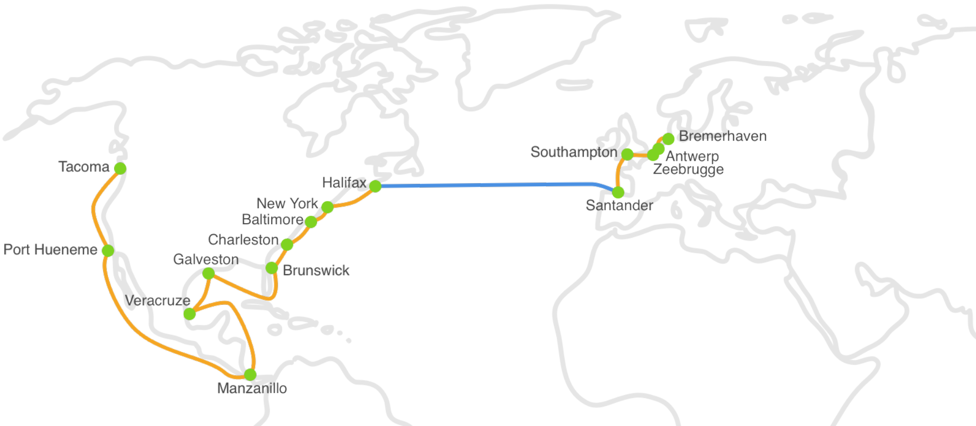


Figure 1.3: A deep sea leg is the section of the trade route over the ocean separating two continents. The deep sea leg portion is marked with a blue line.

Contracts

There are three different type of contracts: Tonnage driving contracts are longer running contracts which drives the need to deploy vessels on given trade routes. If a trade route has a deep sea leg section, tonnage driving contracts are typically loaded before it, and unloaded after sailing it. Way cargo contracts are contracts for transporting goods between two ports that a vessel is already scheduled to visits. These can also be longer running contracts. Spot cargo contracts are one off loads that requires to be transported which can be serviced if a vessel already stops by the corresponding port and has available capacity.

Voyage

Each time a trade route is serviced it is called a voyage. It consists of a sequence of port calls within a trade route. Figure 1.1 shows an example of different voyages in a trade route, indicated by a red and yellow line. Some of the ports in a trade route may have visit restrictions connected to them, while others are free to be visited as seen fit. This means that not all ports in a trade route have to be visited on each voyage.

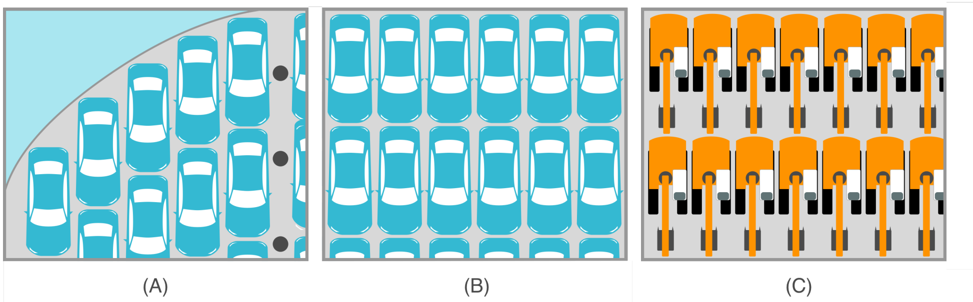


Figure 1.4: (A) Obstruction in vessel holds such as ladders, stanchions or the contours might make it difficult to use space optimally. (B) Cars might be stowed very efficiently with little broken space, because most cars are similar in shape and size. (C) Various heavy construction equipment, in this case excavators, might be irregularly and variably shaped preventing a compact and uniform stow.

Broken space

Broken space is space on a shipping vessel that is lost during stowage of goods. Reasons for broken space might be attributes of the vessel or the shape of the goods to be transported. As an example, see Figure 1.4 below.

1.2 Motivation

This Thesis is a collaboration with one of the world's largest RoRo liner shipping companies, operating a fleet of more than 60 state-of-the-art vessels, capable of carrying a mix of products, from auto and large construction and agriculture machines to various breakbulk products (Logistics, 2016a).

Low demand and excess capacity have caused global shipping rates to fall to a historic low as seen from UNCTAD (2016). Lower margins make it attractive for shipping companies to look for ways to reduce cost. Comparatively little operational research has been done on RoRo shipping, suggesting there is room for improvement. Christiansen, Fagerholt, and Ronen (2004) argue that a good decision support system has the potential for significant benefits to RoRo providers in maritime transportation. UNCTAD (2016) shows how general cargo ships have steadily decreased over the last 40 years. This emphasizes how important improvements are for a RoRo shipping companies competitiveness.

One of the largest contributors to shipping costs is visiting ports, both in regards to time and money. Optimize route planning and keeping port visits to a minimum while fulfilling all contracts within each contract's specification can help to reduce costs. Another aspect where there is room for improvement and cost reduction is in capacity estimation. The cause of costs can be either underestimation of capacity, too much idle space on a vessel which could be used to transport products, or overestimation of capacity, part of the product load left at port causing a breach of contract or delay fees.

1.3 The problem

The Single Trade Routing Problem (STRP) in RoRo shipping is to plan routes for several voyages over a planning horizon, as well as determine which products at which quantities to load and unload at scheduled port visits. The objective is to minimize cost while fulfilling all contracts. The purpose of the work presented in this thesis is to model a version of the STRP with focus on the two causes of cost outline in Section 1.2 to serve as a realistic solution to the problem. The complete solution could help limiting costs as well as serve as a useful decision support tool for a RoRo-shipping company.

1.4 Project outline

This thesis is organized in the following way. In Chapter 2, we review relevant literature. We focus on literature from within the last decade. We give a detailed description of the problem in Chapter 3. Chapter 4 contains the proposed mathematical model for solving the STRP. This chapter includes model assumptions, and representation of the different attributes in the model. The instance generation process is explained in depth in Chapter 5. In Chapter 6, we present results from the computational study. After the computational study, we present an example solution in Chapter 7 where the results of an instance is analyzed in depth. Chapter 8 contains an analysis with the purpose of exploring practical applications of the model.

We conclude the project in Chapter 9 before we make suggestions for future research in Chapter 10.

Chapter 2

Literature Review

In this chapter relevant literature to the Single Trade Routing Problem (STRP) in Roll-on Roll-off liner shipping is presented. The STRP focuses on the tactical level of decisions. The relevant literature presented in this chapter reflects this. Most liner shipping use containers to transport products according to Meng, Wang, Andersson, and Thun (2013). Because of this, much of the literature concerning container shipping is presented to illustrate similar aspects in RoRo shipping.

Even though we aim to present a good overview, the literature review is not an exhaustive and comprehensive presentation of all relevant literature.

Section 2.1 contains general theory and literature about maritime transportation. In Section 2.2, we present the literature and theory on network design, network design options, and examples from the literature on the different network designs. We present literature relevant to cargo routing and scheduling in Section 2.3. Section 2.4 contains literature related to time-dependent attributes, while Section 2.5 presents literature regarding stowage in maritime transportation. In Section 2.6, literature regarding Operations Research in Roll-on Roll-off shipping is presented.

2.1 Maritime transportation

Transportation planning has been widely discussed in the literature, according to Christiansen and Nygreen (1998). Even though transportation has attracted focus from Operations Research (OR), maritime transportation is not the most researched transportation field. Christiansen et al. (2007) gives several explanations for the low attention drawn to the literature by maritime transportation. Low visibility in the global modes of transportation, less structure, and uncertainty regarding weather conditions and mechanical problems as some of these explanations. Even though maritime transportation has been behind in OR, Christiansen et al. (2007) mentions that there has been an increase in research within the field. This is confirmed by the trend Meng et al. (2013) identifies when they present 70 papers that use OR methods to examine containership routing and scheduling problems. 42 of the 70 papers date to 2010 and later.

2.2 Liner network design

Liner shipping, as described above, operates along a given route with a published itinerary in advance. The frequency of each trip might change, but the routes themselves stay the same for years. Because of this, the route network design is an important strategic decision, according to Christiansen et al. (2013). Deciding which ports to visit, in what order and how often, and what type of ship to use at what speed, are all major decisions to make when making the routes (Christiansen et al., 2013). This is the network design problem, deciding the routes and which ports to serve.

The difficulty of solving a network design problem (NDP) was shown by Johnson, Lenstra, and Kan (1978). They proved that solving an NDP is NP-complete, even in the simple case where all arc weights are equal, and the choice is restricted to spanning trees. This was done by reducing an NP-complete 0−1 *knapsack* problem to an NPD. *NP-complete* means the problem is not possible to solve in polynomial time. A *spanning tree* is a sub-graph of an undirected graph connecting all nodes without any cycles in the graph. The same type of proof was performed by Agarwal and Ergun (2008) when they reduced the the $0-1$ *knapsack* problem into a simultaneous ship scheduling and cargo routing problem. Again NP-completeness was shown for the NDP. The complexity of the NDP limits the size of the problems that can be solved using a deterministic model. The typical number of ports in a problem being solved are 10 to 20, as in the Takano and Arai (2011) model, which uses 11 ports and in the Shintani, Imai, Nishimura, and Papadimitriou (2007) problem, which contains 20 ports. They both use a genetic algorithm to solve their NDP.

Different categories of network designs

According to Meng et al. (2013), research on network design can be classified into the four different categories of feeder network, ship routes without transshipment, a hub-and-spoke network, and a general liner shipping network.

Ship routes without transshipment can have pendular routes, where the route goes from one end port to the other end port and back again, or with circular routes. Lu (2002) presented a network design model that is a blend of a circular and a pendular route. Some ports have visiting restrictions on them, forcing them to be visited more than once. The model presented also forces routes to take an integer number of weeks to serve. Chu, Kuo, and Shieh (2003) presented a model for a pendular liner network. This network had weekly departures with an upper limit on the sailing time for a route. Ting and Tzeng (2003) scheduled a circular network. They did this by looking at information regarding time and cost. Shintani et al. (2007) present a model for a pendular line. Speed is one of the aspects that affects the cost in this model. Chuang, Lin, Kung, and Lin (2010) showed how their fuzzy genetic algorithm can generate a good liner route illustrated by a five-port example. Their example also included uncertain demand. Chen and Zeng (2010) researched a network design problem where demand was varying. For their test, they used real geography and artificially created demand. Meng and Wang (2011b) researched a problem where demand was uncertain for the fixed routes.

A general liner shipping network with transshipment is the last classification of container shipping according to Meng et al. (2013). Agarwal and Ergun (2008) presented a model for a liner network design problem with transshipment. The transshipment cost was not taken into account when generating the network. Alvarez (2009) continued on the model proposed by Agarwal and Ergun (2008). They incorporated transshipment cost in the network design. Reinhardt and Pisinger (2012) formulated a model that allowed for transshipment in a specific port. They incorporated transshipment cost in their model.

Even though our problem is not a network design problem, many of the components in modeling the different networks can be used in our STRP. Components that are relevant include visiting restrictions, sailing time limits, and departure restrictions. The network design problem Chen and Zeng (2010) presented contains varying demand and artificially created demand which also is relevant for our problem.

2.3 Cargo routing and scheduling

Cargo routing regards the planning of routing a fleet of vessels. The fleet of vessels is to serve specified cargoes. Scheduling is used when time is included in the planning of the routing. These are usually tactical planning problems.

Jetlund and Karimi (2004) studied a tramp routing and scheduling problem where they obtained the fleet schedule by repeatedly solving for a single vessel. Lin and Liu

(2011) looked into a real tramp ship routing and scheduling problem. The problem contained various types of dry cargo.

It is not always important in long-term contracts to pick up a set amount of cargo each visit. Normally, there is a targeted cargo size with some flexibility. The flexibility can, for example, be ±10% of the targeted amount. G Brønmo, Christiansen, and Nygreen (2007) and Geir Brønmo, Nygreen, and Lysgaard (2010) studied a problem that also includes the need to determine the optimal size of cargo to transport. Korsvik and Fagerholt (2010) also considered a similar problem. Both studies show the improved profit that can come from having the flexibility of cargo to pick up. Hwang, Visoldilokpun, and Rosenberger (2008) reviewed a cargo routing and scheduling problem. All cargo in this issue is assumed to utilize the capacity of a given vessel completely.

Kobayashi and Kubo (2010) studied a routing and scheduling problem where ports are closed for service during nights. This can be viewed as multiple time windows when the planning horizon spans several days. Li and Pang (2011) and Pang, Xu, and Li (2011) studied a problem where ports can only handle one vessel at a time.

Another routing and scheduling problem presents itself when the stowage of cargo also must be considered. Fagerholt, Hvattum, Johnsen, and Korsvik (2013) studied a problem related to this in which they took stowage into account, as well as cargo coupling constraints. Andersson, Duesund, and Fagerholt (2011b) presented a problem where they considered synchronization of the delivery of some cargoes.

One constraint in most of the ship routing and scheduling problems is that cargo being picked up cannot be serviced by more than one vessel. Andersson, Christiansen, and Fagerholt (2011a) and Korsvik, Fagerholt, and Laporte (2011) studied a problem where cargo can be split between multiple vessels. Both papers show that by splitting cargo, utilization of the fleet can be improved, thus improving the profit as well. Hennig, Nygreen, Furman, Song, and Kocis (2011) proposed a formulation for routing and scheduling, where one of the aspects differed from the two studies mentioned before. They proposed a quantity requirement for loading and unloading ports within a certain time window. The quantity can be split between multiple vessels.

Cargo routing and scheduling literature focus on cargo, which is relevant for us. Relevant aspects include the flexibility in how much cargo each vessel has to pick up at each visit, optimal cargo to transport, visiting restrictions in the form of time and handling, synchronization of some cargo, and splitting of cargo between vessels.

2.4 Time-dependent attributes

Speed

One trend that has brought emphasis on speed optimization in recent years is the steep increase in bunker fuel price (Christiansen et al., 2013). According to Papadakis and Perakis (1989), 50% of total operating costs come from fuel consumption. This makes speed important when trying to minimize operation cost. With an increase in speed, time spent sailing will go down at the same time as bunker fuel consumption increases. The opposite happens with a decrease in speed. To keep cost down at the same time as fulfilled contracts according to transit time restrictions requires a certain speed. Bunker consumption changes depending on speed as shown by A. Perakis and Jaramillo (1991). There are different takes on what the relationship between bunker consumption and vessel speed is. Ronen (2011) and Yao, Ng, and Lee (2012) use a cubic convex function while Fagerholt, Laporte, and Norstad (2010) use a quadratic convex function. Regardless of the relationship between speed and bunker consumption, it is a non-linear function.

Norstad, Fagerholt, and Laporte (2011) present a model where speed optimization is incorporated by having an upper and lower bound for speed. This makes a non-linear factor in the objective function as well as in time-related restrictions. Andersson, Fagerholt, and Hobbesland (2015) propose a new linearized modeling approach for integrating a speed optimization when planning ship routes. They use a linear combination of predefined discrete speed alternatives in order to provide desired speed.

We will use Andersson et al. (2015) approach of using a linear combination of predefined speed alternatives in order to incorporate speed optimization and use Yao et al. (2012) take on bunker fuel consumption to calculate operating cost for vessels.

Evenly spread

Evenly spread restrictions can be implemented as both hard and soft constraints. A hard constraint set conditions for the variables that are required to be satisfied. With a soft constraint, variables are penalized in the objective function for not being satisfied.

Belanger, Desaulniers, Soumis, and Desrosiers (2006) study a periodic airline fleet deployment problem. The goal in this problem is to generate a schedule that can be repeated. When the departure of two flights with the same route is scheduled too closely, a penalty cost is added to the objective function. This is a good example of a soft constraint.

Sigurd, Ulstein, Nygreen, and Ryan (2005) include time separation requirements on recurring visits to the same port in their general pickup and delivery problem. They are doing this by generating predefined patterns with time separation requirements included. The mathematical model is restricted to only choose one of these patterns for each customer. Halvorsen-Weare, Fagerholt, Nonås, and Asbjørnslett (2012) include spread of departures by the same principle as Sigurd et al. (2005) in their supply vessel planning problem. They do not make patterns in advance but include one restriction for each group of number-of-visits-in-a-week. Both of these examples with evenly spread restrictions requires that the number of ports to visit during the planning horizon is known in advance.

Norstad, Fagerholt, Hvattum, Arnulf, and Bjørkli (2015) have another take on evenly spread where they include a parameter that determines the minimum accepted time between two consecutive voyages on a trade route. Again the number of voyages that have to be performed on the trade route is known in advance.

The problem we are trying to solve does not operate with a fixed number of voyages.

Transit time
Transit time restrictions make sure products being transported have a lower transshipment time than the maximum transit time set. Implementing transit time restrictions in a maritime transportation problem is important to mimic real world problems. Gelareh, Nickel, and Pisinger (2010) presents a model for a hub and spoke network where the demand between two ports are dependent on the transit time, as well as the price of shipping.

Álvarez (2012) and S. Wang and Meng (2011) use transit time for cargo that is transported to deal with level of service. Álvarez (2012) claims that the level of service is not properly addressed by cost minimization. He use a linear function of the cargoes transit time through the liner shipping network in order to address the level of service. S. Wang and Meng (2011) are also looking at the transit time of cargo that is being transported in order to explain the level of service.

2.5 Stowage in maritime transportation

From Table 1.2, we saw that container ships take up a larger share than RoRo vessels, which is a subset of general cargo ships, of the world fleet. This helps explain why the majority of research done on stowage in maritime transportation is on container ships. In container stowage, the objective is to minimize the time it takes to load and unload, (Ambrosino, Sciomachen, and Tanfani, 2004). It can also be to minimize the number of container movements, (Avriel, Penn, Shpirer, and Witteboon, 1998).

Øvstebø, Hvattum, and Fagerholt (2011a) analyzed stowage plans for RoRo ships with hoistable decks. They used vehicle transportation vessels as examples. Later Øvstebø, Hvattum, and Fagerholt (2011b) extends their problem from stowage on a fixed route to a routing problem with stowage constraints. The routing problem they solved was to find a pickup and delivery route for one vessel. The problem included mandatory ports and optional ports for the vessel to visit. We have not found any other papers dealing with stowage on RoRo vessels.

2.6 Use of operations research in roll-on roll-off shipping

Roll-on Roll-off (RoRo) shipping is not as well discussed in the literature as in other segments. This is because RoRo shipping is a minor section of liner shipping. What follows is a selection of Operations Research done in RoRo shipping the last decade.

Sigurd et al. (2005) presented a network design problem for a set of routes between several ports. The vessels that were serving the network were fast RoRo vessels. Fagerholt, Johnsen, and Lindstad (2009) looked into the fleet deployment problem in RoRo shipping. They modeled voyages on a liner route with starting time windows. Andersson et al. (2015) proposed a new model for planning shipping routes in which optimization of ship speed along the routes was integrated into the model. It is common to use a sequential approach where speed first is given, and then later optimize it. This deployment and routing problem is studied using RoRo shipping.

Kang, Choi, Kim, and Park (2012) and Jung et al. (2011) presented a decision support system for a car carrier with the objective to determine the number of cars each vessel should pick up on each route, all while minimizing cost. Cars not serviced in one planning horizon adds an extra cost to the objective function. Lindstad, Asbjørnslett, and Strømman (2011) analyze carbon dioxide (CO_2) emissions based on, among others, the RoRo segment in the world fleet.

Patricksson, Fagerholt, and Rakke (2015) proposed a model of the maritime fleet renewal problem, which also included emission regulations as a limitation. They used RoRo shipping as their case study.

Fischer et al. (2016) presented different strategies to include robustness, as well as different strategies for handling disruption when assigning a fleet of vessels to predefined voyages while trying to minimize cost in RoRo liner shipping.

The STRP in RoRo shipping has not yet been researched, to the best of our knowledge.

Fagerholt et al. (2009) looked into the fleet deployment problem, and Sigurd et al. (2005) present a network design problem. We plan routes for several voyages over a planning horizon, as well as which products at which quantity to load and unload at scheduled port visits. One important, and to the best of our knowledge, new aspect of this problem is estimating how much space products will actually use when stored, including broken space.

Chapter 3

Problem Description

In this chapter, the Single Trade Routing Problem (STRP) in RoRo shipping is presented. After the STRP is presented, a small example of the problem is given.

The central components of the STRP are: The planning horizon, trade routes, voyages, ports, vessels, and contracts. The planning horizon is the overall time window for the problem. A typical length of the planning horizon is one to two months. A trade route is a logistical network identified as a set of pathways and stoppages for commercial transport. Stoppages are typically ports, and will be referred to as such from now on. A vessels transit from one origin port to a destination port is called a voyage. A voyage over a trade route may include stops at other ports between origin and destination. Vessels are the means of transportation used on voyages.

Vessels may have different stowage capacities, both in area and volume. Vessels in RoRo shipping typically come with hoistable decks to accommodate for differences in product sizes. Hence vessel capacity, both in area and volume, can be adjusted to accommodate different needs. The different decks may have different weight restrictions, limiting what can be stowed there, as a function of the quantity and weight of products. The vessels can travel at different speeds depending on scheduling needs. Speed variations affect the fuel consumption of vessels, and thus also affect the cost. The speeds at which a vessel can travel may also vary with vessel type.

Which ports to visit and what should be transported during a voyage are determined by which contracts will be handled during the planning horizon. A contract defines demand for products, that is, which products to be transported at which quantities. Limits on inventory capacity and production rates impose limits on the quantity of products, which can be available for transport at given times. Hence, a contract may require pickups to be fairly evenly spread in time across the planning horizon. A contract may have pickup frequency restrictions, that is, it may require that the total quantities to be transported be divided over several or all voyages during the planning horizon. A contract may also define an upper and lower bound on the quantities which have to be transported during the voyage the contract is handled. A contract defines at what port to pick up the products, and which port to transport them to, as well as restrictions on transit time. The transit time restrictions dictate within what time interval a contract has to be fulfilled.

Products are grouped into product types. A Product type is a class of products that have some attributes that are similar, such as size or functionality. An example of two different product types are cars and heavy construction equipment. Different product types may need to be stowed in different ways. Large and heavy product types, like heavy construction equipment, may need to be securely tethered, which creates broken space around it. Hence, different product types might have different degrees of space utilization, that is, have a different typical broken space to stowage space ratio. This makes it difficult to decide product type mix and quantities that can be stowed during a port visit.

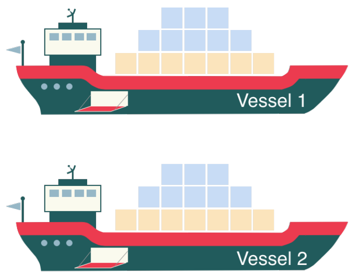
The objective of the STRP is to find routes for all planned voyages during a planning horizon that minimizes costs while fulfilling all contracts. The decisions to be made are which ports to visit during a given voyage, and what quantities of the different product types should be loaded and unloaded at each of these port visits.

3.1 Example

The following is a simplified example of the STRP. We are continuing the same example as used in Vallestad and Weggersen (2016). New considerations are taken into account. The input for this example is as follows: A trade route consisting of three ports, two vessels with different capacities, four contracts, and two different types of products.

The assumptions are that it is only possible to make two voyages. Each contract only asks for one product type, time is measured in days, and all ports have equal visiting cost. The only attribute differentiating the vessels is capacity. It is also assumed that both vessels have an equal bunker fuel consumption to sail speed ratio.

There are two different product types as stated above. Product type one (P_1) and Product type two (P_2). All products of the same type have the same need for area and volume. P_2 has a bigger need for area and volume than P_1. There are different decks on the vessels that accommodate for the various product types. Deck attributes



	Capacity
Product type 1	100
Product type 2	50
Product type 1	75
Product type 2	30

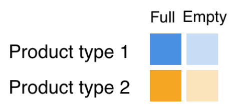
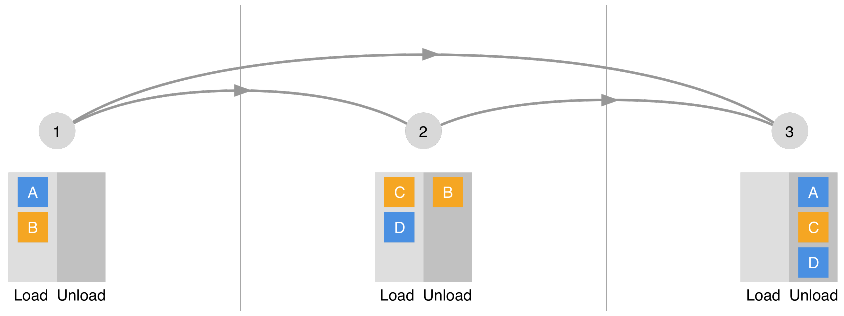


Figure 3.1: The different vessels with corresponding capacities

can limit what kind of product type can be placed on it. Decks set up for P_1 can only have P_1 since it has the lowest need of area and volume. Decks set up for P_2, on the other hand, can also stow P_1.

Figure 3.1 shows the vessels used in this example and their capacities. Vessel one (K_1) and Vessel two (K_2). The capacity of K_1 is 100 of P_1 and 50 of P_2. For K_2, the capacity is 75 of P_1 and 30 of P_2.



	A	B	C	D
Product type 1	75	0	0	120
Product type 2	0	30	30	0
Transit time (days)	4	2	1.8	2

Figure 3.2: The trade route with all contracts

There are four contracts used in this example. These contracts are seen in Figure 3.2. The color of the contracts matches the type of product since they can only hold one type. The table in the bottom right corner shows each contract demand, as well as transit time restrictions. Transit time restrictions are being explained in more depth later in the example. Contracts A (C_A) and D (C_D) have demand for P_1, while contracts B (C_B) and C (C_C) have demand for P_2.

The objective in this example is to fulfill all contract demands using two voyages. The cost of visiting a port and the distance between ports are the only two elements that affect the objective.

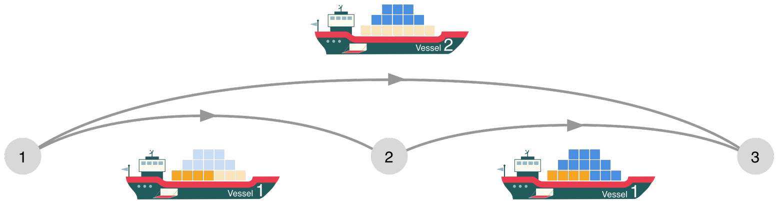
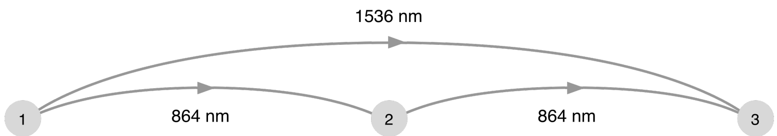


Figure 3.3: A solution of the example problem using two voyages

Figure 3.3 shows a solution to this example. One voyage is sailed by V_1 from port 1 to 3, through 2, to handle C_B, C_C, and C_D. The demand of P_2 in C_B is less than V_1's capacity of the same product type so the demand of C_B can fit on V_1 without any problems. When V_1 arrives at port 2, it unloads P_2 from C_B before loading on the P_1 and P_2 from C_D and C_C respectively. P_2 from C_C has no problem being placed at its allocated space on the vessel. The amount of P_2 from C_C is less than V_1's capacity for that of P_2. C_D demands an amount of P_1 that exceeds the vessels capacity for that product type. Since V_1 only loaded 30 of its capacity of P_2 and can load 50 of it, it has 20 extra spaces that P_1 can use. The combined capacity of P_1 and what is left of P_2 on V_1 is enough to accommodate for C_D.

V_2 sails the second voyage directly from 1 to 3 to hand C_A and has no problem loading the demand of P_1 from C_A on board. V_2 saves the cost of visiting port 2 and the time it takes to do so.



	nm/day	fuel/day (tones)	fuel/nm (tones)	864 nm (days)	1536 nm (days)
16 knots	384	43	0.112	2.3	4
18 knots	432	57	0.133	2	3.6
20 knots	480	75	0.157	1.8	3.2

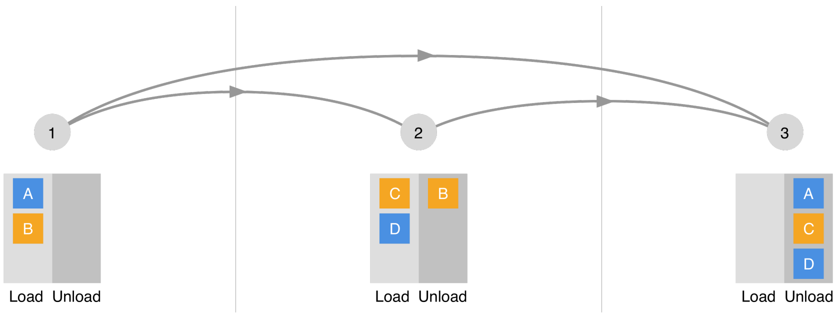
Figure 3.4: Distances and bunker fuel consumption dependent on vessel speed

The contracts that are being handled in this example also contains transit time restrictions as seen in Figure 3.2. C_A is handled by V_2 when being sailed directly from port one to port three and has a transit time of four days. Figure 3.4 shows that it takes four days to sail between the two ports with a speed of 16 knots. 18 knots and 20 knots would also satisfy the transit time constraint. V_2 sails at 16 knots in order to minimize fuel consumption and cost.

V_1 sails between port one and two with a speed of 18 knots. If it had sailed at 16 knots, the transit time restriction would not be satisfied of C_B. When V_1 sails between port two and three, C_C and C_D are loaded on board. This last route can not be sailed with anything less than 20 knots, even though C_D allows V_1 to sail the route with 18 knots. This is because of the transit time restriction in C_C.

Evenly spread

Now lets assumed that C_C has a product demand of P_2 equal to 60 instead of 30 as shown in Figure 3.5, and everything else stay the same as over.



	A	B	C	D
Product type 1	75	0	0	120
Product type 2	0	30	60	0
Transit time (days)	4	2	1.8	2

Figure 3.5: Distances and bunker fuel consumption dependent on vessel speed for evenly spread

This would mean that one vessel could not handle C_C on its own due to the capacity restrictions of each vessel. K_1 sail the same route as before, picking up C_B from port one, and from port two, C_D and half of C_C. Since K_1 can not handle C_C on its own, K_2 has to visit port two as well to pick up half of P_2 demand of C_C.

To satisfy the evenly spread requirements of contract C_C, the two voyages can not start at the same time. To simplify the example, the planning horizon is six days. One solution is to let K_1 start at day zero and let V_2 start on day two. This makes the first visit in port two to handle C_C at day two and the second visit at day four. If the route to K_1 and K_2 had not been the same, the starting time could be different. It is not when a vessel starts a voyage that is important, but that certain contracts are handled according to spread restrictions.

In order to further show the depth of evenly spread, just one port with four contracts that have to be handled at this port is looked at. This can be seen in Figure 3.6. The contracts have a different number of times they have to be visited, as well as when each visit should be made. It is very unlikely that all contracts can be handled at their desired service times. By minimizing the sum of all delays, a good spread can still be found.

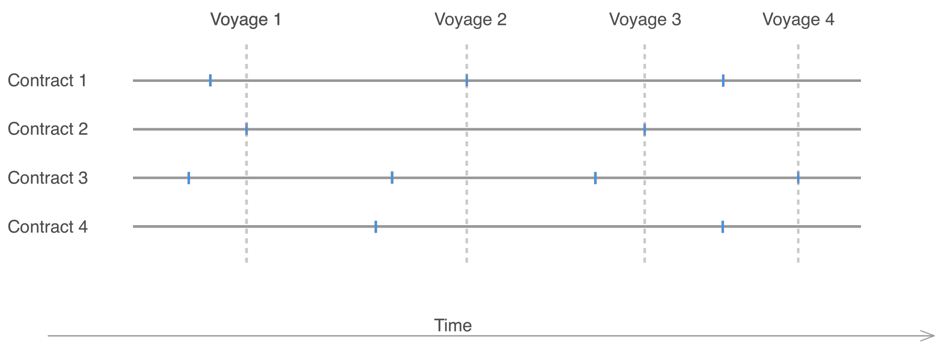


Figure 3.6: Four contracts that have to be handled in one port where visits have been done, so that an even spread can be fulfilled

Figure 3.6, conveys that Voyage one is handling Contracts one, two, and three. Even though Contract one is the only one on Voyage one being handled at its desired time, the two other contracts are being handled shortly after their desired times. The same can be seen in Voyage two, three, and four. The result of doing it this way is that an even spread is achieved.

Chapter 4

Mathematical Model

In this chapter we propose a base model for solving the STRP. Section 4.1 contains our modeling assumptions regarding demand and supply, transportation, the fleet, load planning, and time. In Section 4.2, the trade route network representation is described as well as the connection between the networks and contracts, along with bunker fuel consumption, contract representation, stowage capacity constraints on vessels, space utilization of different product types, and evenly spread of contracts. In the final section, Section 4.3, a presentation of the mathematical model for the STRP is given.

4.1 Model assumptions and limitations

Demand and supply
All contractual demands are known with certainty in advance of the planning horizon. Supply is constant and the quantities to be transported are always available when needed at all ports during the entire planning horizon. Partial satisfaction of demand is not allowed.

Transport
Vessels moves unidirectionally through the trade route, from lower indexed ports to higher indexed ports. Transshipment of demand is not accommodated, and the operating costs at port are known and fixed. The cost of sailing is determined by bunker fuel consumption. The fuel price is fixed while bunker fuel consumption vary from vessel to vessel.

Fleet
The fleet composition in this problem is heterogeneous with a finite number of different types of vessels.

Transport logistics
The problem involves stowing quantities of products of different product types on vessels. Each product has a base area. We assume a linear relationship between the products base area, and the area that product will use of vessel capacity when stowed on board. That is, its base area plus any broken space created by stowing the product. We further assume that the broken space to base area ratio is approximately equal for products of same product type, at least on average. In summation, we assume given the total base area x_p of all products of each product type p, total real space use S can be predicted with a tolerable error from Equation 8.1,

$$S = F_1 x_1 + ... + F_p x_p, \tag{4.1}$$

and that the space utilization factors $F_1, ..., F_p$ can be estimated. In this project we do not take weights or volume into account, nor varying ship capacity caused by hoistable decks.

4.2 Model description

4.2.1 Trade route network

We use network representation for our single trade routes. Origin, destination and contract ports are collectively termed nodes, and the transportation links, or routes, between two ports are termed arcs. The weight of the arc between two given nodes is the distance which needs to be traveled in order to get from one node to the other.

In the example in Figure 4.1, it is assumed that our vessel begins its voyage in node one. This simplification makes formulating a model and generating data simpler, but at the same time it makes the model much less useful. Artificial origin and destination nodes are added to our network to include information about the origin and destination points of each vessel. One origin node $o(k)$ and one destination node $d(k)$ is added for each vessel. This can be seen in Figure 4.2. For each origin node, arcs between it and every other node except other origin nodes are included in the network. For each destination node, arcs between it and its corresponding origin node and every port node, except other destination nodes, are included in the network. All arcs are assigned weights corresponding to the direct travel distance between the two

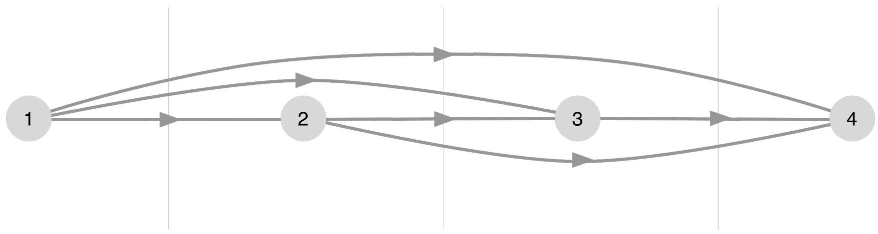


Figure 4.1: Illustrates a network representation of a trade route with four ports. Note that the arcs in this example network is unidirectional, limiting a vessel to only move forward during its voyage. In this example, it is assumed that the vessels starts in node one.

ports which the arc belong to. Figure 4.2 shows an updated version of the network in Figure 4.1, with origin and destination nodes. Vessels are still limited to travel unidirectional through the network, but they are no longer limited to start in a port of the trade route. This allows us to modell a larger set of problem instances. For example, some vessels may now start in a destination port from its previous voyage, which is not necessarily a part of the trade route for its next voyage.

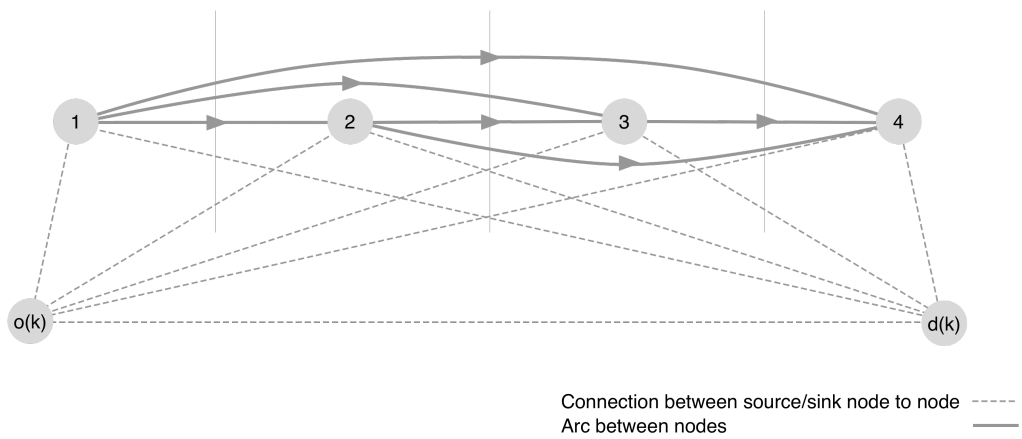


Figure 4.2: The same network as in Figure 4.1, with artificial origin and destination nodes.

4.2.2 Bunker fuel consumption

Bunker consumption is a major part of shipping costs. If a vessel sails at a low speed, the fuel consumption is low and as such, cost will be low as well. If a vessel sails at a high speed, the argument is the opposite. The function representing the speed against bunker consumption is convex function as seen in Section 2.4.

To linearize bunker consumption in our model, we use a piece-wise linear approximation of the sailing cost function as presented in Andersson et al. (2015). The speed is found by using a linear combination, as shown in Equation 4.2, of two different consecutive speed alternatives, s_i. The linear combination can describe any speed alternative V between the two speed alternatives used. To find the bunker fuel consuption corresponding to a specific speed, the corresponding values of w are used to calculate bunker cost from Equation 4.3.

$$V = w_{ijvks} \cdot s_i + (1 - w_{ijvks}) \cdot s_{i+1}, \qquad w_{ijvks} \in [0,1],\ i \in \{1, 2, ..., n-1\} \tag{4.2}$$

$$F_V = w_{ijvks} \cdot f_i + (1 - w_{ijvks}) \cdot f_{i+1}, \qquad w_{ijvks} \in [0,1],\ i \in \{1, 2, ..., n-1\} \tag{4.3}$$

4.2.3 Contract

A contract is a binding agreement between a shipping company and a client who requires goods to be transported between a loading port and an unloading port. The contract stipulates a demand, a quantity of goods to be transported which might be of different product types. The contract must also stipulate pickup frequency bounds, that is the minimum and maximum number of voyages the contractual demand could be divided over. For example, if both minimum value and maximum value is set to two, the contractual demand should be divided over exactly two voyages. In addition, the contract must also stipulate minimum and maximum quantity which can be loaded during a given pickup. For example, the lower load quantity bound can be set to zero, and the upper load quantity bound can be set to the total demand, which would put no restrictions on how the total demand should be divided up over the required amount of voyages. A contract must also stipulate a transit time requirement, which ensures that the contract will be completed within a given time from pickup, and might require that the pickups for one contract should be fairly evenly spread in time. Similar to the load quantity bounds, the transit time might be defined to put no completion time restrictions on a contract.

Connecting contracts and networks

Each port node in a trade route network has two sets assign to it, one set for collecting contracts to be loaded at that port, the loading set, and one set for collecting contracts which should be unloaded at that port, the unloading set. These can be seen visualized in 4.3. Each contract is placed in the loading set corresponding to its loading port and in the unloading set corresponding to its unloading port. The index functions l(c) and u(c) returns loading port and unloading port respectively for a given contract c. An example of this can be seen in Figure 4.3, where a small instance with four ports and three contracts has been illustrated.

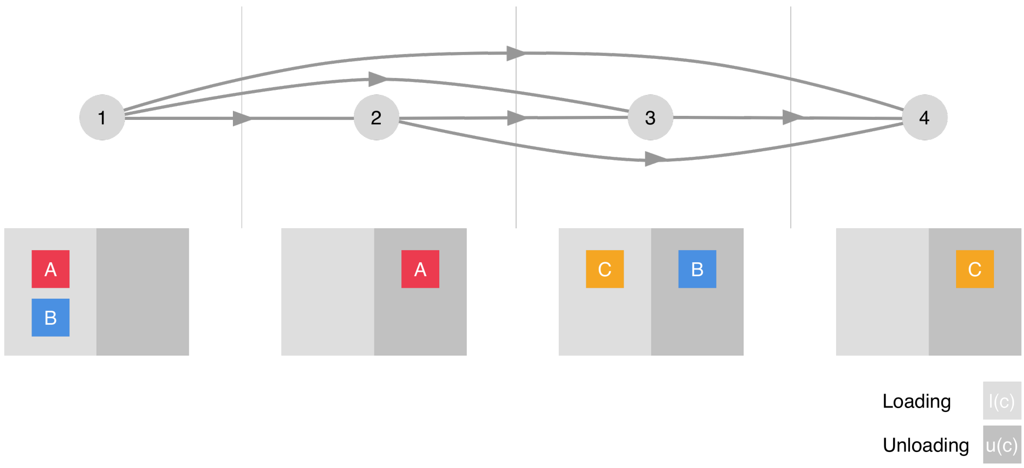


Figure 4.3: Loading contracts and unloading contracts for a small instance with four ports and three contracts.

4.2.4 Stowage capacity

As outlined in the problem in Chapter 3, a vessel may have different types of decks with differences in capacity attributes. Any good to be transported may be limited by its size attributes to only fit on a subset of these decks on a given vessel. As an example, consider a vessel with three different decks with different storage capacity attributes, height, weight and base area. The vessel transports goods which can roughly be divided into three product types by its height, weight and base area, product types A, B, and C. Goods of product type A are low and light, relative to decks capacity attributes, and may be stored on all decks. Product type B are higher and heavier, and may only be stored on two of the decks. Product type C are higher and very heavy, and may only be stored on one of the decks. This is illustrated in Figure 4.4 below.

The important thing to note in the Figure 4.4 is that the vessels storage capacity is not the sum of the height of the three columns. Rather, it is the sum of the height of the three thin lines to the left in the figure. Each column is the maximum storage capacity for a given product type, and does not necessarily represent a fixed space

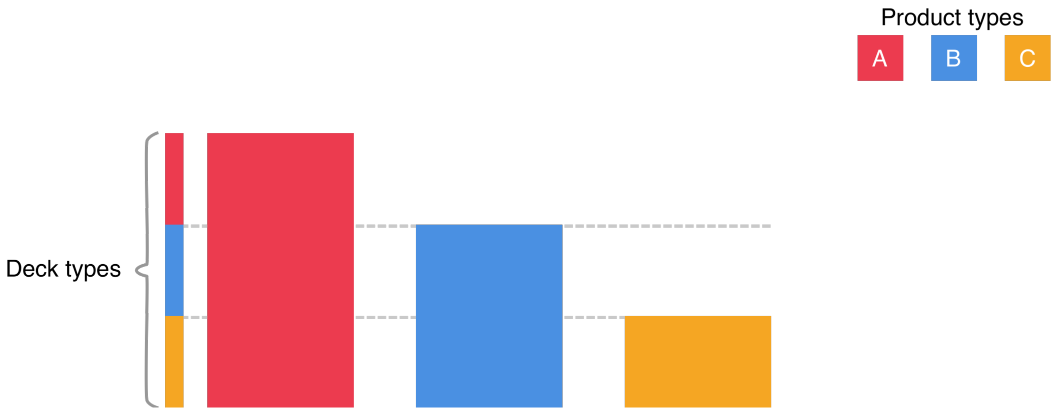


Figure 4.4: Different product types.

available only for that product type.

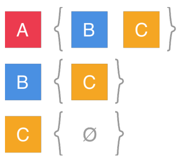


Figure 4.5: Different product type sets.

Each product type X has a set assigned to it, an overlap set P_p^S, containing the product types which storage space could also be used to store product type X. In our example above, for A, this set would be B, C. The rest of the overlap sets from this example is shown in Figure 4.5. In Figure 4.6, is an example of a inventory instance during a voyage, while transporting goods between port nodes one and two. Product type A could be cars, B could be trucks, and C could be heavy construction equipment. All storage capacity on the deck available for storing cars has been used. The storage capacity on deck two has been used for storing both cars and trucks. The storage capacity for deck three has been used to store heavy construction equipment. Both deck two and three has open, unused space. To find the actual space available for stowing cars in this instance, you would have to subtract the stowed amount of trucks and the stowed amount of heavy construction equipment from that maximum storage capacity for cars.

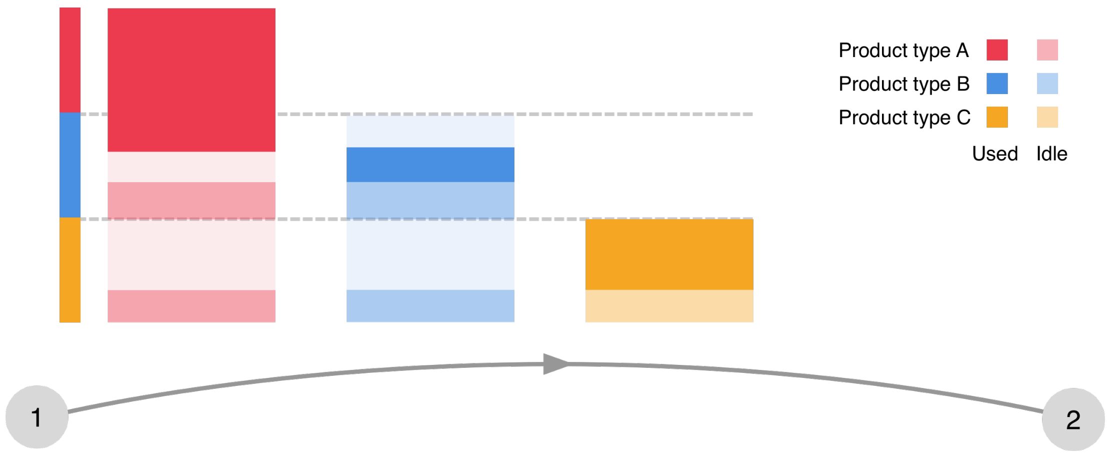


Figure 4.6: Products transported between ports.

4.2.5 Space utilization

Each deck type might store goods with different degree of space utilization, that is, typical broken space to stowed space ratio. As an example, let us consider a deck type which typically stores excavators. This has been illustrated in Figure 4.7. An excavator has a square base, but because of the boom, dipperstick and bucket they use space less efficiently than for example a typical car. A model which does not take this into account, but calculates required storage space from the square base of the excavators will underestimate the required space. By calculating the ratio between the total area actually used to store the excavators and the total base area of the excavators we have a space utilization factor (SUF) which can be used to estimate required space.

A deck type is typically not limited to store a specific kind of good, like an excavator, but to product types which are similar in spatial attributes. Rather than calculating an exact SUF, and average SUF for specific product types can be estimated from historical data. The SUF can then be used to adjust the space requirement of a demanded quantity to reflect the actual space needed to stowage the quantity, based on which product type it belongs to. The estimation of these factors are beyond this study, but a possible method would be multiple linear regression, or relying on the subjective estimates of people with experience.

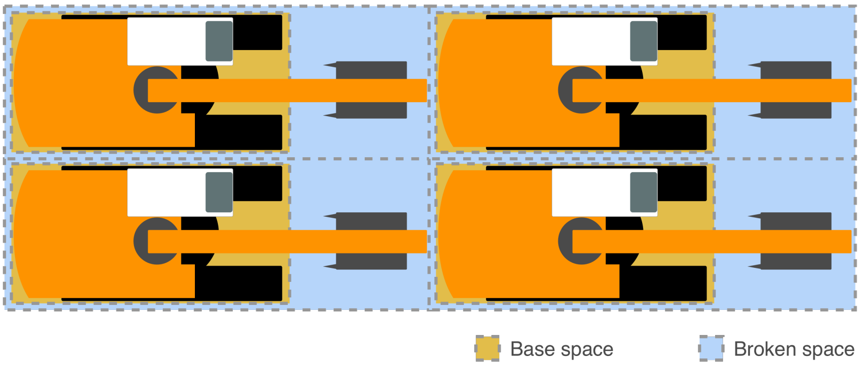


Figure 4.7: By calculating the ratio between the total area actually used to store the excavators and the total base area of the excavators we can calculate the space utilization factor for the stowage configuration in the figure. In this case the space utilization factor is 1.67.

4.2.6 Fairly even spread

As mentioned in Section 4.2.3, a contract might require its pickups to be fairly evenly spread throughout the planning horizon. A commonly used method to ensure fairly even spread is to define time window for each pickup, which the pickup should occur during. This requires that the contract requires an exact number of pickups for a planning horizon in advance. If the number of pickups can be chosen from an interval however we cannot with certainty know in advance how many pickups a given contracts demand will be divided over. This makes it difficult to use time windows for this case while keeping the model linear.

Instead of using time windows, we have implemented fairly even spread by using separation requirements. Each contract which require fairly evenly spread are required to spread it pickups with a separation time which falls within an interval calculated with Equation 4.4,

$$\frac{T^h}{b_c} \pm s_c \tag{4.4}$$

where T^H is the planning horizon length, b_c is the number of pickups chosen by the model, and s_c is acceptable slack. The model allows deviations from this requirement by adding a penalty cost in the objective function that depends on the contract and the size of the deviation.

As an example, if a contract is serviced two times and planning horizon window is 28 days, the optimal spread between the two pick is 14 days. Let us assume the first pickup is done at day 3 while the second pickup is done at day 13 with a spread of 10 days. With a slack of ± 2 days the maximum spread is 16 days and the minimum spread is 12 days, which makes this pickup 2 days to early. This contract would now increase the total cost by the contract penalty cost multiplied by 2. However, if the second pickup is done at time 16, making the spread 13 days, this would be within our acceptance interval.

4.3 The mathematical model

The model presented in this chapter is based on the work done by Vallestad and Weggersen (2016). The model has been extended in several ways: we have added time constraints, constraints to spread contract pickups fairly evenly in time and enforce transit time requirements for contracts, and added constraints which allows for variable vessel speed. In addition, we have extended the capacity constraints from a simple fixed capacity for each deck type, which can store a specific product type, to allow loading of a product type on any deck type which has the appropriate attributes.

The given problem can be formulated on a directed graph $G = (\mathcal{N}', \mathcal{A})$ where the set $\mathcal{N}'$ contains all nodes, and $\mathcal{A}$ is the set of arcs. The set $\mathcal{N}$ contains the nodes representing ports. Let $\mathcal{K}$ denote the set of all vessels in the fleet of the shipping company. Each $k \in \mathcal{K}$ has an node $o(k) \in \mathcal{N}'$ representing the initial position for k, and an artificial node $d(k) \in \mathcal{N}'$ representing its final destination. The set $\mathcal{N}'$ is as such, $\mathcal{N}' \subseteq \mathcal{N} \cup \{o(k), d(k)\}$. Each vessel k is available at its initial position $o(k)$ at time T_k^A. The set $\mathcal{V}$ contains the different voyages v of the given planning horizon. A voyage v for a vessel k corresponds to a path from $o(k)$ to $d(k)$ in G.

Let $\mathcal{N}_k \subseteq \mathcal{N} \cup \{o(k), d(k)\}$ be the subset of nodes, where each vessel k is compatible with nodes $\mathcal{N}_k$. Let $\mathcal{A}_k \subset \mathcal{N}_k \times \mathcal{N}_k$ be the set of arcs (i, j) such that the vessel can sail directly from the port associated with node i to the port associated with node j. Sailing from node i to node j using vessel k at speed s incurs a variable cost C_{ijks}^S and total sailing time T_{ijks}^S. Each k visiting port i incurs a cost C_i^V and time T_{ik}^P.

The set $\mathcal{P}$ contains the different product types p that can be handled by vessels, while the set $\mathcal{P}_p^S$ contains the different product types that can be stowed on the same deck as product type p. The maximum capacity of a given product p on vessel k is given by K_{kp}^{CAP}, and each vessel k has a space utilization factor U_{kp} for each product type p.

Let $\mathcal{C}$ denote the set of all contracts. Each contract $c \in \mathcal{C}$ is associated with a loading node $l(c)$ and an unloading node $u(c)$ as well as a maximum transit time T_c^T. The set of contracts with products to be loaded in a port i is denoted by $\mathcal{C}_i^L$, while the set of contracts with products that should be unloaded in port i is denoted by $\mathcal{C}_i^U$.

Each contract c has a minimum and a maximum number of times the contract has to be serviced given by $[\underline{P}_c, \bar{P}_c]$. Each contract c has a minimum and maximum load quantity for each product type p, $[\underline{Q}_{cp}, \bar{Q}_{cp}]$ on each visit. The quantity loaded of product type p during one port visit to partly or fully meet a contract demand has to be within these bounds. The total demand of each product p for a contract c is denoted by D_{cp}.

A contract c that has its pickup node C_i^L before the deep sea leg, is confined by the evenly spread restriction. The cost of breaking the evenly spread restriction for contract c is given by C_c^P.

The continuous time variable t_{iv} represents the time at which a port visit begins at node i if served on voyage v. The time a vessel k is available is given by T_k^A, while T_{ik}^P describes the time spent in port i for vessel k. The total length of planning horizon is given by T^H.

The binary variable δ_{cvk} is equal to 1 iff products for contract c is being loaded on a voyage v with a vessel k. Continuous variables l_{ijvp} indicate the total load on board vessel k after completing service at node i during voyage v, and the variables q_{cpvk} indicate the loaded quantity of product type p from contract c on vessel k sailing voyage v.

Several networks are implemented with binary flow variable, x_{ijvk}, which is equal to 1 iff a vessel k on a voyage v sails directly between the nodes i and j, and 0 otherwise. The set $\mathcal{S}$ contain the discrete speed alternatives s ordered from low to high. The variable w_{ijvks} takes values between $[0, 1]$ depending on what linear combination of the speed alternatives s vessel k on voyage v has between node i and j. The sum of all w_{ijvks} for each arc in use is 1.

The variables β_c give the number of times contract c is serviced during a planning horizon, while ϕ_{nc} is equal to 1 iff contract c is visited n times. ϕ_{nc} is defined for the interval $[\underline{P}_c, \bar{P}_c]$. The binary variable z_{cvw} is 1 iff voyage v and w are evenly spread for contract c when $w > v$. When contract c is not evenly spread between two voyages, the continues variables s_c give the number of days the evenly spread restriction is violated.

The problem can then be formulated as follows:

Sets and indices

$\mathcal{N}'$	Set of all nodes, i
$\mathcal{N}$	Set of all ports, i
$\mathcal{V}$	Set of voyages, v
$\mathcal{A}$	Set of all arcs
$\mathcal{A}_k$	Set of arcs ,(i, j), for vessel k
$\mathcal{K}$	Set of vessels, k
$\mathcal{C}$	Set of all contracts, c
$\mathcal{C}_i^L$	Subset containing loading contracts in a given port i

$\mathcal{C}_i^U$	Subset containing unloading contracts in a given port i
$\mathcal{P}$	Set of product types, p
$\mathcal{P}_p^S$	Set of product types which can be stored in the same spaces as product type p
$\mathcal{S}$	Set of speed alternatives s, ordered from low to high

Parameters

C_{ijks}^S	Cost of sailing for vessel k from node i to j at speed s
C_i^V	Cost of visiting port i
C_c^P	The penalty cost of deviating one day from evenly spread on contract c
D_{cp}	Total demand of product type p in contract c
K_{kp}^{CAP}	Capacity of product type p on vessel k
U_{kp}	Space utilization factor of product p on vessel k
$\underline{Q}_{cp}$	Minimum quantity to load from contract c of product p
$\bar{Q}_{cp}$	Maximum quantity to load from contract c of product p
$\underline{P}_c$	Minimum number of times contract c has to be serviced
$\bar{P}_c$	Maximum number of times contract c has to be serviced
T_{ijks}^S	Time spent sailing with vessel k at speed s from node i to j
T_c^T	Maximum transit time for contract c
T_k^A	Time vessel k is available at origin node
T_{ik}^P	Time vessel k spends in port i
T^H	Length of planning horizon

Variables

$$x_{ijvk} \quad \begin{cases} 1, & \text{if voyage } v \text{ use vessel } k \text{ to sails from node } i \text{ to } j \\ 0, & \text{otherwise} \end{cases}$$

$$\delta_{cvk} \quad \begin{cases} 1, & \text{if contract } c \text{ is handled by vessel } k \text{ on voyage } v \\ 0, & \text{otherwise} \end{cases}$$

$$z_{cvw} \quad \begin{cases} 1, & \text{if voyage } v \text{ and } w \text{ are evenly spread for contract } c, \text{ w > v} \\ 0, & \text{otherwise} \end{cases}$$

ϕ_{nc}	$\begin{cases} 1, & \text{if contract } c \text{ is visited } n \text{ times. Defined for } \underline{P}_c \leq n \leq \bar{P}_c \\ 0, & \text{otherwise} \end{cases}$
l_{ijvp}	Quantity of product type p stored on the vessel chosen for voyage v when the vessel departs from node i to sail for node j
q_{cvpk}	Quantity of product p loaded on vessel k used on voyage v for contract c
t_{iv}	Time starting service at node i on voyage v
w_{ijvks}	The weight of speed alternative s when sailing from node i to node j with vessel k on voyage v
β_c	Number of times contract c is serviced during a planning horizon
s_c	The number of days of slack needed to service contract c with evenly spread restrictions

Objective function

$$\begin{aligned} min\ z = & \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \mathcal{A}_k} \sum_{v \in \mathcal{V}} \sum_{s \in \mathcal{S}} C^S_{ijks} w_{ijvks} + \\ & \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \mathcal{A}_k} \sum_{v \in \mathcal{V}} C^V_i x_{ijvk} + \sum_{c \in \mathcal{C}} C^D_c s_c \end{aligned} \tag{4.5}$$

The objective function (4.5) sum all variable costs associated with sailing the fleet and visiting ports, in addition to adding a penalty cost for violating the evenly spread constraints.

Network constraints

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{N} \cup d(k)} x_{o(k)jvk} = 1, \qquad v \in \mathcal{V} \tag{4.6}$$

$$\sum_{i \in \mathcal{N}} x_{ijvk} - \sum_{i \in \mathcal{N}} x_{jivk} = 0, \qquad k \in \mathcal{K},\ v \in \mathcal{V},\ j \in \mathcal{N}' \tag{4.7}$$

$$\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N} \cup o(k)} x_{id(k)vk} = 1 \qquad v \in \mathcal{V} \tag{4.8}$$

$$\sum_{i \in \mathcal{N}'} x_{ijvk} \leq 1 \qquad k \in \mathcal{K},\ v \in \mathcal{V},\ j \in \mathcal{N}' \tag{4.9}$$

Constraints (4.6) - (4.8) are flow constraints. They ensure that each voyage starts in a origin node, that every visited node is also exited, and that each voyage ends up in a destination node. Constraints (4.9) make sure each port can only be visited at most once each voyage.

Speed constraints

$$x_{ijvk} = \sum_{s \in \mathcal{S}} w_{ijvks}, \qquad k \in \mathcal{K},\ (i,j) \in \mathcal{A}_k,\ v \in \mathcal{V} \tag{4.10}$$

Constraints (4.10) describe the relationship between the flow variables and the speed-dependent variables, such that the weights of the speed alternatives add up to 1 if the vessel k sails between node i and j, 0 otherwise.

Vessel constraints

$$\sum_{j \in \mathcal{N}} \sum_{v \in \mathcal{V}} x_{o(k)jvk} \leq 1, \qquad k \in \mathcal{K} \tag{4.11}$$

Constraints (4.11) make sure that each vessel is only able to be used once for each planning horizon.

Capacity constraints

$$\sum_{j \in \mathcal{N}} l_{jivp} + \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}_i^L} U_{kp} q_{cvpk} - \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}_i^U} U_{kp} q_{cvpk} - \sum_{j \in \mathcal{N}} l_{ijvp} = 0, \quad i \in \mathcal{N},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.12}$$

$$0 \leq l_{ijvp} \leq \sum_{k \in \mathcal{K}} K_{kp}^{CAP} x_{ijvk} - \sum_{p' \in \mathcal{P}_p^S} l_{ijvp'}, \quad (i,j) \in \mathcal{A},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.13}$$

$$\sum_{j \in \mathcal{N}} l_{o(k)jvp} = 0, \qquad k \in \mathcal{K},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.14}$$

Constraints (4.12) require that for each product, vessel class, and voyage, the sum of product quantity entering a node plus loaded product quantity and minus the unloaded product quantity must be equal to the product quantity leaving the node. Constraints (4.13) state that the quantity of product type p, carried on voyage v, by vessel k, over an arc from node i to node j must not be greater than the total vessel capacity for that product, subtracting space which could be used for this product type, but has already been used by goods of other product types which can use the same space. In addition, it states that if an arc is not in use, vessel capacity is equal to zero. Constraints (4.14) ensure that each voyage starts with an empty vessel.

Pickup constraints

$$\underline{P}_c \leq \sum_{v \in \mathcal{V}} \sum_{k \in \mathcal{K}} \delta_{cvk} \leq \bar{P}_c, \qquad c \in \mathcal{C} \tag{4.15}$$

$$\delta_{cvk} \leq \sum_{i \in \mathcal{N} \cup o(k)} x_{il(c)vk}, \qquad k \in \mathcal{K},\ c \in \mathcal{C},\ v \in \mathcal{V} \tag{4.16}$$

$$\delta_{cvk} \leq \sum_{i \in \mathcal{N} \cup o(k)} x_{iu(c)vk}, \qquad k \in \mathcal{K},\ c \in \mathcal{C},\ v \in \mathcal{V} \tag{4.17}$$

Constraints (4.15) ensure that upper and lower bound for contract pick frequency c is respected. Constraints (4.16) and (4.17) state that for a contract c to be handled on a voyage v, the vessel for that voyage has to visit both the loading and unloading port for that contract.

Loading constraints

$$\underline{Q}_{cp} \delta_{cvk} \leq q_{cvpk} \leq \bar{Q}_{cp} \delta_{cvk}, \qquad k \in \mathcal{K},\ c \in \mathcal{C},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.18}$$

$$\sum_{k \in \mathcal{K}} \sum_{v \in \mathcal{V}} q_{cvpk} = D_{cp}, \qquad c \in \mathcal{C},\ p \in \mathcal{P} \tag{4.19}$$

Constraints (4.18) ensure that upper and lower bound for loading for each contract and product type is respected, and forces loading to be equal to zero if a contract is not handled on a voyage v. Constraints (4.19) ensure that the demand for each product type p required by each contract c is satisfied during the planning horizon.

Time constraints

$$t_{o(k)v} \geq T_k^A \qquad k \in \mathcal{K},\ v \in \mathcal{V} \tag{4.20}$$

$$x_{ijvk}(t_{iv} + \sum_{s \in \mathcal{S}} T_{ijks}^S w_{ijvks} + T_{ik}^P - t_{jv}) \leq 0 \qquad k \in \mathcal{K},\ (i,j) \in \mathcal{A}_k,\ v \in \mathcal{V} \tag{4.21}$$

$$\sum_{k \in \mathcal{K}} \delta_{cvk}(t_{l(c)v} + T_c^T - t_{u(c)v}) \geq 0 \qquad v \in \mathcal{V},\ c \in \mathcal{C} \tag{4.22}$$

$$t_{iv} + \sum_{j \in \mathcal{N}} \sum_{s \in \mathcal{S}} T_{ijks}^S w_{ijvks} \leq T^H, \qquad v \in \mathcal{V},\ i \in \mathcal{N}^{\mathcal{O}} \tag{4.23}$$

Constraints (4.20) ensure that a voyage v to be sailed with a vessel k cannot begin before that vessel is made available at its origin node. Constraints (4.21) describe the compatibility between routes and schedules. The time for start of service in port j cannot be less than the sum of the start time of service in port i, the service time for loading and unloading, and the sailing time from port i to port j with vessel k, if vessel k is really sailing between port i and j. Constraints (4.22) ensure that if a contract is being handled on a voyage by a vessel, δ_{cvk} equal to 1, the time it takes to sail from the loading port to the unloading port on that voyage is less than the transit time restriction T_c^T. Constraints (4.23) ensure that no voyage is startet after the defined planning horizon.

Evenly spread constraints

$$\beta_c = \sum_{v \in \mathcal{V}} \sum_{k \in \mathcal{K}} \delta_{cvk}, \qquad c \in \mathcal{C} \tag{4.24}$$

$$z_{cvw}[\frac{T^H}{\beta_c} - s_c - (t_{l(c)v} - t_{l(c)w})] \leq 0, \qquad c \in \mathcal{C},\ v \in \mathcal{V},\ w \in \mathcal{V}|w > v \tag{4.25}$$

$$z_{cvw}[\frac{T^H}{\beta_c} + s_c - (t_{l(c)v} - t_{l(c)w})] \geq 0, \qquad c \in \mathcal{C},\ v \in \mathcal{V},\ w \in \mathcal{V}|w > v \tag{4.26}$$

$$\sum_{v \in \mathcal{V}} \sum_{w \in \mathcal{V}|w>v} z_{cvw} \geq \beta_c - 1, \qquad c \in \mathcal{C} \tag{4.27}$$

$$\sum_{w \in \mathcal{V}|w>v} z_{c1w} \leq 1, \qquad c \in \mathcal{C} \tag{4.28}$$

$$\sum_{w \in \mathcal{V}|w>v} z_{cw|\mathcal{V}|} \leq 1, \qquad c \in \mathcal{C} \tag{4.29}$$

$$\sum_{w \in \mathcal{V}|w>v} z_{cvw} = \sum_{w \in \mathcal{V}|w>v} z_{cwv}, \qquad c \in \mathcal{C},\ v \in \mathcal{V}|v \neq 1,\ v \neq |\mathcal{V}| \tag{4.30}$$

Constraints (4.24) sets the value of he variable β_c equal to the amount of pick ups during the entire planning horizon for that contract. Constraints (4.25), (4.26) and (4.27) forces contract pickups on different voyages to be spread with a time difference, or spread value, equal to the planning horizon divided by the amount of pick ups that contract has during the planning horizon, plus the value of the slack variable s_c, which can be increased at a penalty cost. Constraints (4.28) and (4.29) forces the first and last voyage to be, respectively, the first and last voyage in time. Constraints (4.30) ensure that contracts on voyages are compared with the same contract on the next voyage in time, ensuring that two or more voyages cannot be compared to to the same voyage.

Binary and non negativity constraints

$$x_{ijvk} \in \{0,1\}, \qquad k \in \mathcal{K},\ (i,j) \in \mathcal{A}_k,\ v \in \mathcal{V} \tag{4.31}$$

$$\delta_{cvk} \in \{0,1\}, \qquad k \in \mathcal{K},\ c \in \mathcal{C},\ v \in \mathcal{V} \tag{4.32}$$

$$z_{cvw} \in \{0,1\}, \qquad c \in \mathcal{C},\ v \in \mathcal{V},\ w \in \mathcal{V}|w > v \tag{4.33}$$

$$l_{ijvp} \geq 0, \qquad i \in \mathcal{N},\ j \in \mathcal{N},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.34}$$

$$q_{cvpk} \geq 0, \qquad k \in \mathcal{K},\ c \in \mathcal{C},\ v \in \mathcal{V},\ p \in \mathcal{P} \tag{4.35}$$

$$t_{iv} \geq 0, \qquad i \in \mathcal{N},\ v \in \mathcal{V} \tag{4.36}$$

$$\beta_c \geq 0, \qquad c \in \mathcal{C} \tag{4.37}$$

$$s_c \geq 0, \qquad c \in \mathcal{C} \tag{4.38}$$

$$0 \leq w_{ijvks} \leq 1, \qquad k \in \mathcal{K},\ (i,j) \in \mathcal{A}_k,\ v \in \mathcal{V},\ s \in \mathcal{S} \tag{4.39}$$

(4.31) - (4.33) are binary constraints for x_{ijvk}, δ_{cvk} and z_{cvw} while (4.34) - (4.38) are non negativity constraints for l_{ijvp}, q_{cvpk}, t_{jv}, β_c and s_c. Constraints (4.39) make sure w_{ijvks} only can take values between 0 and 1.

Linearization of Constraints (4.21) and (4.22)

$$t_{iv} + \sum_{s \in \mathcal{S}} T^S_{ijks} w_{ijvks} + T^P_{jk} \leq t_{jv} + M^T_{ijv}(1 - x_{ijvk}), \quad k \in \mathcal{K},\ (i,j) \in \mathcal{A}_k,\ v \in \mathcal{V} \tag{4.40}$$

$$t_{l(c)v} + T^G_c + M^C_c(1 - \delta_{cvk}) \geq t_{u(c)v}, \quad v \in \mathcal{V},\ c \in \mathcal{C} \tag{4.41}$$

The value of M^T_{ijv} is set to the maximum time it takes to sail between port i and j on voyage v. The maximum time includes sailing time between the two ports sailing at the lowest speed and port visit times. If the ports to visit do not have consecutive port numbers, every port is being visited and visiting times at all the ports are included in the maximum time, while sailing at the lowest speed. This is lower than what the value would have been if there was only one M. The value of M would have been set to T^H. This is because $(t_{iv} + \sum_{s \in \mathcal{S}} T^S_{ijks} w_{ijvks} + T^P_{jk} - t_{jv})$ will never be larger than T^H. The lowest M could be set to is $T^H - min(T_{ijks})$. If M was set to anything lower than $T^H - min(T_{ijks})$, it would be possible for a voyage to include port visits even though the voyage might not be needed. Since voyages might have different $min(T_{ijks})$, it would not be possible to set one value that would fit every case.

M_c is set to the time it takes to sail from $l(c)$ to $u(c)$ including visiting every port in between using the lowest speed possible, and subtracting T^T_c of contract c.

Linearization of Constraints (4.25) and (4.26)

$$\sum_{i \in \mathcal{V}} i\phi_{ic} = \sum_{v \in \mathcal{V}} \sum_{k \in \mathcal{K}} \delta_{cvk}, \qquad c \in \mathcal{C} \tag{4.42}$$

$$\sum_{i \in \mathcal{V}} \phi_{ic} = 1, \qquad c \in \mathcal{C} \tag{4.43}$$

$$\sum_{i \in \mathcal{V}} \frac{T^H}{i} \phi_{ic} - s_c - T^H(1 - z_{cvw}) \leq t_{l(c)v} - t_{l(c)w}, \quad c \in \mathcal{C},\ v \in \mathcal{V},\ w \in \mathcal{V} | w > v \tag{4.44}$$

$$\sum_{i \in \mathcal{V}} \frac{T^H}{i} \phi_{ic} + s_c + T^H(1 - z_{cvw}) \geq t_{l(c)v} - t_{l(c)w}, \quad c \in \mathcal{C},\ v \in \mathcal{V},\ w \in \mathcal{V} | w > v \tag{4.45}$$

$$\phi_{nc} \in \{0, 1\}, \qquad c \in \mathcal{C},\ n \in \mathcal{N} \tag{4.46}$$

Constraints (4.42) and (4.43) ensure that a binary variable ϕ_{ic} with the index i equal to the value of β_c is equal to 1 while ϕ_{ic} for index i not equal to the value of β_c is equal to zero, for each contract. ϕ_{ic} is then used in constraints (4.44) and (4.45) to get the correct spread value without having to divide T^H over the variable β_c, which makes the model non-linear. (4.46) are binary constraints for ϕ_{nc}.

Chapter 5

Instance Generation

In this chapter we describe our instance generator in detail, outlining which choices were made and why. The aim is to create instances describes realistic enough approximations of real planning horizons to make the computational study in Chapter 6 meaningful. In chapter 4, we outlined a mathematical approximation of the Single Trade Routing Problem. Understanding the instance generating process will be helpful when analyzing our solutions to see how well the problem relates to our model.

In Section 5.1, the input data is presented. Section 5.2 contains a detailed description of the instance generator, and how it uses the input data to generate the remaining necessary data. In section 5.3 we present a reference scheme for the test instances, while the last section, Section 5.4, contains an overview over all test instances used throughout the computational study.

5.1 Generator input data

In this section we present the input data, which are trade routes used to generate distance data and network topology, maximum port pickup frequencies, which sets upper and lower bounds on the random generation of contract pickup frequency, and bunker fuel consumption rates and capacity ratios, which are used to generate vessels. Some of the data provided by the company is proprietary, and in those cases the example data shown has been fabricated.

Trade routes

A trade route is a logistical network of stoppages and pathways used for the commercial transport of goods. A trade route is represented here by a distance matrix, which contains all distances between all stoppages, here ports, which it is possible to sail directly between. We chose three trade routes sailed by the case company, with different level of ports, and created distance matrices for them by using an online tool for calculating sailing distances between the ports (Sea-Distance, 2017.)

An illustration of the trade routes with the corresponding port distance matrix representation can be seen below. Figure 5.1 illustrates a trade route with five ports, with America as its origin and Asia as its destination. Figure 5.2 illustrates a trade route with ten ports, with Asia as its origin and Europe as its destination. Figure 5.2 illustrates a trade route with 15 ports, with Europe as its origin and America as its destination. The corresponding port distance matrices can be seen in Table 5.1, Table 5.2 and Table 5.3.

Figure 5.1: The America to Asia trade route

	Baltimore	Manzanilo	Port Hueneme	Tacoma	Yokohama
Baltimore	0	1904	4926	5968	9634
Manzanilo		0	3022	4064	7730
Port Hueneme			0	1069	4770
Tacoma				0	4268
Yokohama					0

Table 5.1: Distances between ports in the America to Asia trade route. All distances are in nautical mile (nm)

Figure 5.2: The Asia to Europe trade route

	Yokohama	Kobe	Shanghai	Laem Chabang	Singapore	Alexandria	Piraeus	Southampton	Antwerp	Bremerhaven
Yokohama	0	353	1036	2932	2892	8062	8499	10960	11185	11385
Kobe		0	783	2733	2691	7861	8298	10759	10984	11184
Shanghai			0	2204	2237	7407	7844	10305	10530	10730
Laem Chabang				0	784	5954	6391	8852	9077	9277
Singapore					0	5170	5607	8068	8293	8493
Alexandria						0	512	2939	3164	3364
Piraeus							0	2622	2847	3047
Southampton								0	257	457
Antwerp									0	357
Bremerhaven										0

Table 5.2: Distances between ports in the Asia to Europe trade route. All distances are in nautical mile (nm)

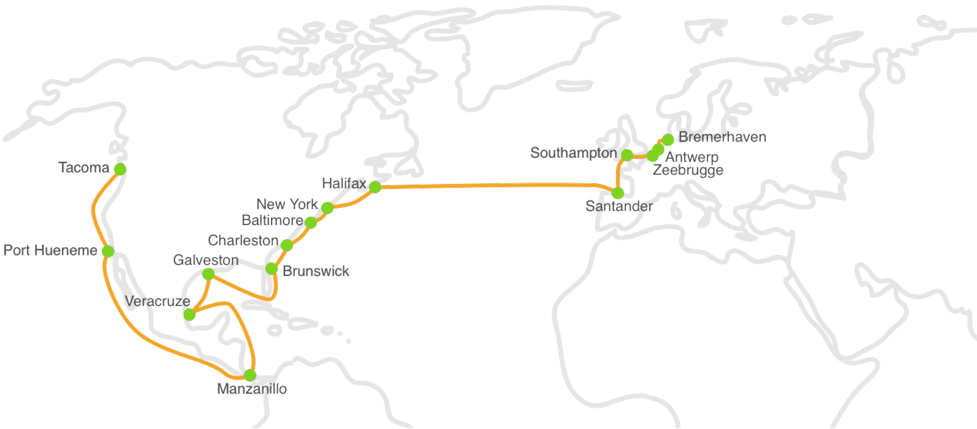


Figure 5.3: The Europe to America trade route

	Gothenburg	Bremerhaven	Zeebrugge	Southamton	Santander	Halifax	New York	Baltimore	Charleston	Brunswick	Galveston	VeraCruz	Manzanillo	Port Hueneme	Tacoma
Gothenburg	0	373	544	696	1192	2923	3568	3855	3999	4157	5167	5262	5037	7988	9101
Bremerhaven		0	292	457	953	2938	3583	3870	4014	4172	5182	5277	5006	7957	9070
Zeebrugge			0	186	682	2716	3317	3604	3748	3949	4943	5038	4735	7686	8799
Southamton				0	528	2555	3156	3443	3587	3788	4782	4877	4574	7525	8638
Santander					0	2545	3109	3396	3558	3711	4705	4800	4466	7379	8492
Halifax						0	593	899	1077	1240	2303	2399	2298	5249	6362
New York							0	410	629	791	1862	1958	1972	4923	6036
Baltimore								0	552	715	1785	1880	1904	4855	5968
Charleston									0	199	1336	1430	1560	4511	5624
Brunswick										0	1226	1321	1505	4456	5569
Galveston											0	622	1485	4436	5549
VeraCruz												0	1410	4361	5474
Manzanillo													0	2951	4064
Port Hueneme														0	1165
Tacoma															0

Table 5.3: Distances between ports in the Europe to America trade route. All distances are in nautical mile (nm)

Maximum port pickup frequencies

Experimenting with contract generation, we learned that the method used for generating contract pickup frequencies has a large effect on model solutions. For example, let us consider randomly generating upper and lower bounds for contract pickup frequencies from an interval between and including zero and the number of voyages for a planning horizon. If the number of contracts is larger enough, there is a high probability this would lead to there being at least one contract for each port with a contract pickup frequency of four. Using this method would lead to almost all solutions having the vessel visit every port on every voyage, and because of this, we chose to go for a method which gave to more realistic results. When an instance is created with a given trade route, maximum port pickup frequencies for that trade route is also provided. For example, the maximum port pickup frequencies for a trade route with ten ports, used for creating a planning horizon with four voyages might look something like this,

$$4,\ 3,\ 4,\ 3,\ 2,\ 2,\ 3,\ 4,\ 4,\ 3$$

In this case, contracts generated with port five as a loading port, can at most have a contract pickup frequency upper bound of two. These maximum contract pickup frequencies were created using port visit data from our case company. These were used to rank ports based on importance.

Bunker fuel consumption rates

The bunker fuel consumption rate for a given speed alternative is looked up in tables similar to Table 5.4. These tables contain the bunker fuel consumption rate for all speed alternatives. We have chosen to use $300 per ton as the bunker fuel consumption cost, since this was the price of bunker fuel at the moment the Thesis was written, Bunker (2017). This might not reflect the bunker fuel cost in the future.

		RoRo vessel								
	Speed	12	12.5	13	13.5	14	14.5	15	15.5	16
Fuel	ton/nm	0.100	0.105	0.109	0.114	0.119	0.124	0.129	0.134	0.140
	ton/day	28.898	31.386	34.033	36.849	39.843	43.025	46.406	49.997	53.811
	Speed	16.5	17	17.5	18	18.5	19	19.5	20	20.5
Fuel	ton/nm	0.146	0.152	0.159	0.166	0.173	0.180	0.188	0.196	0.204
	ton/day	57.858	62.153	66.709	71.540	76.662	82.090	87.843	93.936	100.390

Table 5.4: Fabricated fuel consumption rates for different speed alternatives for one RoRo vessel

Capacity Ratios

Our case company has contracts with clients which produce goods with different height and weight capacity requirements. An example of this is cars and heavy construction equipment. To accommodate these clients, our case company has vessels with different decks designated for different product types. The demand for some product types may be larger than for others, so total capacity might be divided unevenly among these decks. In addition, some product types might be stored on deck types designated for other product types. When generating vessels for our test instances, we use capacity ratios together with a vessels total capacity to find the available capacity for product types. For example, a vessel has decks designated for product types A, B, and C. The capacity ratio for product type A is 100% as goods of this product type can be stored anywhere on the ship. Goods of product types B and C can only be stored on subsets of the decks, and has a capacity ratio of 70% and 20% respectively.

5.2 Instance generator

In this Section we describe the three conceptual modules which together make up the instance generator. In the first Section, Section 5.2.1, we describe the Vessel pool module that generates a fleet of vessels from which we sample vessels when generating instances. In Section 5.2.2, all network related data is generated. In the last Section 5.2.3 we describe the Contract module, which generates contracts for instances. Unless stated otherwise, all random values are sampled from a uniform distribution.

5.2.1 Vessel pool module

The case company has a fleet of vessels which belongs to a relatively small set of vessel types. This fleet has a fixed number of vessels and composition over small periods of time. To simulate this, the module generates a pool of vessels, belonging to a small number of vessel classes, which are then used as a common set of vessels to sample from when generating instances. The module takes as input:

- *The number of vessels to be generated*
- *The number of deck types each vessel will have*
- *The number of distinct vessel classes*
- *The bunker fuel consumption rates for each distinct vessel class*
- *The capacity ratios for each distinct vessel class*

The module divides the number of vessels to be created over the number of classes specified in the input. For example, if the module is tasked with creating eight vessels and four different vessel classes, each class will have two vessels. Each class is assigned a set of bunker fuel consumption rates as well as a set of capacity ratios. The module divides the interval of possible vessel sizes into smaller, equal sub intervals, one for each class. The class with the lowest bunker fuel consumption rates gets its size randomly chosen from the first sub interval, the class with the second lowest bunker fuel consumption rates gets its size randomly chosen from the second size sub interval, and so on. This is illustrated in Figure 5.4. Once a vessel has been provided with a total quantity, the module calculates how much of this area should be used for the specific deck types using the provided capacity ratios.

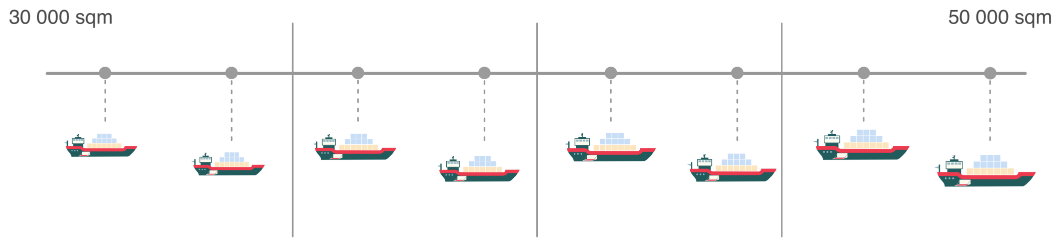


Figure 5.4: Four classes of vessels are created, provided with four different sets of bunker fuel consumption rates. Each class has two vessels, creating a total of eight vessels.

This module also randomly creates space utilization factor (SUF) for each deck type on each vessel. To simulate that larger, heavier goods usually creates more broken space, the module generates increasing SUFs for each deck type, with the first deck type starting with a SUF of 1.0, and then increasing for each additional deck type, using Algorithm 1. The algorithm works as follows: the first product type for each vessel always has an SUF equal to one. The remaining product types draw their SUF from a moving interval, to make sure that the SUF is larger than the last product types SUF. The α-variable can be adjusted to increase or limit the spread between the SUFs. For our vessel pool α is set to 1.3.

Algorithm 1 generateSpaceUtilizationFactors()

$Vessels \leftarrow$ All vessels
$ProductTypes \leftarrow$ All contracts
$SpaceUtilizationFactors \leftarrow$ empty dictionary

for $k \in Vessels$ **do**
  for $p \in ProductTypes$ **do**
    if $p == 1$ **then**
      $SpaceUtilizationFactors(k, p) \leftarrow 1.0$
    else
      $SpaceUtilizationFactors(k, p) \leftarrow 1.0 + Uniform(\frac{p-1}{10}, \frac{p}{10})^{\alpha}$
return $SpaceUtilizationFactors$

Figure 5.5: The Algorithm for randomly creating space utilization factors (SUFs) for each product type, and therefore also for the corresponding deck type, for every vessel.

5.2.2 Trade route module

This module generates all network related parameters, such as sailing costs, sailing time, and available ves sels. The module takes as input:

- *A trade route*
- *A vessel pool*
- *The number of vessels $\mathcal{K}$ available for the planning horizon*
- *The maximum number of voyages $\mathcal{V}$ which will be performed during the planning horizon*

Vessels that are available for a given trade route will differ for different planning horizon. To simulate this, the module samples $\mathcal{K}$ vessels from the vessel pool.

The given trade route is represented by a port distance matrix, which contains the distances between all ports which it is possible to sail directly between, as detailed in Section 5.1. For each possible combination of ports i and j, and vessel k and speed alternative s the module calculates a sailing cost and sailing time. Sailing cost is calculated by multiplying the bunker fuel cost rate corresponding to speed alternative s with the distance between ports i and j. The bunker fuel cost rate is found in the bunker fuel cost rates array corresponding to vessel k. Sailing time is calculated by Equation 5.1, by dividing the distance between port i and j over speed alternative s.

$$T^S_{ijks} = d_{ij}/v_{ks}, \tag{5.1}$$

5.2.3 Contract module

This module generates contracts, and takes as input:

- *The number of contracts to be serviced during the planning horizon*
- *The maximum contract pickup frequencies*

The module generates two different types of contract: tonnage driving contracts, and way cargo contracts. Tonnage driving contracts are longer running contracts which drives the need to deploy vessels to given trade routes. Way cargo contracts are contracts for transporting goods between two ports that the we are already planning to visit. Our instance generator approximates these two types of contracts by only applying fairly evenly spreading requirements to the tonnage driving contracts. In addition, tonnage driving contracts are generated to be loaded before the deep sea leg of a voyage and unloaded afterwards, while way cargo contracts has no such restrictions. A contract is designated to be tonnage driving with a probability of γ, and waywards with a probability of $1 - \gamma$. After instance feasibility evaluation, γ was set to 70%. A loading port and an unloading port are randomly selected and assigned to each generated contract from a set of possible port combination, which depends on contract type. In addition the set of possible combinations is limited further by the restrictions that loading ports are to precede unloading ports, and that the last port can not be a loading port.

The maximum port pickup frequencies are used to randomly generate maximum and minimum pickup frequencies for each contract, with the restriction that minimum pickup frequency should be less than or equal to maximum pickup frequency, and that the maximum pickup frequency should be less than or equal to the maximum port pickup frequency for the contracts loading port.

A contract between a client and the shipping company specifies the quantity of goods, the contract demand, that the shipping company is required to transport. The module generates the total demand for the planning horizon, by calculating the average of all available vessels capacities and multiplying it by the number of voyages that will be performed during the planning horizon. This is again multiplied with a factor r^{CD}, the demand to capacity ratio, which can be adjusted to increase or decrease the demand. Contracts have different loading and unloading ports, meaning that all contracts are not necessarily serviced at the same time. This allows for the total demand to be adjusted above approximate total vessel capacity without causing infeasibility issues. According to the case company, a typical r^{CD} value would be around 95%. In our test instances, we therefore used the r^{CD} values 85%, 95% and 105%. Average capacity is dependent on which vessels are sampled for the planning horizon. If the number of sampled vessels exactly equal the number of voyages to be performed, an r^{CD}-value of 105% would make the total demand exactly 5% larger than total available capacity. When sampling a number of vessels larger than the number of voyages this number is more of an indication, and the real r^{CD}-value depends on which vessels are actually used for the planning horizon.

To vary the size of contract demand, the total demand is divided unevenly over all contracts based on contract pickup frequencies. Contracts which can have more pickups will have a larger proportion of the total demand. Each contract c will have a demand for each product type p called the contract product type demand, or D^C_{cp}. The total contract demand is divided over the different product types p based on the average proportion designated for that product type on all available vessels.

The model has restrictions for maximum and minimum load quantity for a contract during a single pickup. The module calculate these bounds for each product type with Equation 5.2 and 5.3. The load quantity lower bound $\underline{Q}_{cp}$ for a product type p is calculated by dividing the contract product type demand over the maximum number of pickups. The load quantity upper bound $\overline{Q}_{cp}$ for a product type p is calculated by dividing the contract product type demand over the minimum number of pickups. The β value is varied to change the level of variation allowed in load quantity for pickups. For our test instances β was set to 10%.

$$\underline{Q}_{cp} = (1 - \beta) \cdot \frac{D^C_{cp}}{\overline{P}_c} \tag{5.2}$$

$$\overline{Q}_{cp} = (1 + \beta) \cdot \frac{D^C_{cp}}{\underline{P}_c} \tag{5.3}$$

A contract can be defined with restrictions on transit times, which requires unloading to take place within less than or equal time to the specified transit time after loading. These transit times are generated by finding the minimum sailing time $\underline{T}_{ij}$ and maximum sailing time $\overline{T}_{ij}$ between the loading port i and unloading port j as in Equation 5.4,

$$T_c^T = \underline{T}_{ij} + (\overline{T}_{ij} - \underline{T}_{ij}) \cdot Uniform(0,1) + T_c^S \tag{5.4}$$

where T_c^S is an accepted contract specific slack time. For our test instances T_c^S was set to four.

A contract can require that its pickups should be fairly evenly spread in time. Rather then randomly assigning this requirement to contracts, the module assigns it to contracts with loading ports before the deep sea leg of the trade route require that pickups are fairly evenly spread. Contracts after the deep sea leg of the trade route are way-wards contracts, which usually does not require fairly evenly spread.

We have learned that contracts with larger demand are usually considered more important. Hence, if the model has to chose between prioritizing a small contract and a large contract, it should prioritize the large contract. In addition, some clients might pay more for having their pickups evenly spread, regardless of the order size. To simulate this, the cost for deviating from the fairly evenly spread requirement is calculated by multiplying the total demand of the contract by two factors: a quantity unit penalty cost, $\hat{C}_c^P$, and a random prioritization factor, F^P. The unit penalty cost is constant for all contracts, and affects how an instance should prioritize evenly spread requirements relative to the objective of minimizing sailing and port visits costs. The factor F^P is 1 with a probability of 75%, 2 with a probability of 20%, and 3 with a probability of 5%. In most cases the contracts with larger demand will be considered more important and are prioritized for evenly spread when there is a scheduling conflict, unless a smaller contract has randomly been assigned higher priority by F^P. The calculation of penalty cost for deviation, C_c^P, for a contract c is summarized in Equation 5.5.

$$C_c^P = F^P \cdot \hat{C}^P \cdot \textstyle\sum_{p \in P} D_{cp} \tag{5.5}$$

5.3 Instance reference scheme

In order to clearly differentiate between instances, they are named according to the seven dimensions that varies:

1. *The trade route*
2. *The number of ports*
3. *The number of contracts*
4. *The number of voyages*
5. *The number of available vessels*
6. *The demand to capacity ratio*
7. *The unit penalty cost for evenly spread deviation*

An instance group is named by concatenating the dimensions listed above, and separating them by underscores. Trade routes are abbreviated in the following way: America to Asia (AmAs), Asia to Europe (AsE), and Europe to America (EAm). Table 5.5 shows the reference scheme as well as the domain of the dimensions.

As an example, a test instance with the following dimension values, Asia to Europe with 10 ports, 100 contracts, 4 voyages, 8 available vessels, a demand to capacity ratio equal to 95% belong to the group AsE_10_100_4_8_95.

Instance reference scheme		
Group of instances	TR_P_C_V_K_DCR	
Single instance	TR_P_C_V_K_DCR_I	
Where	Trade route (TR)	$\in \{AmAs, AsE, EAm\}$
	Number of Ports (P)	$\in [5, 15]$
	Number of Contracts (C)	$\in [50, 300]$
	Number of Voyages (V)	$\in [4, 8]$
	Number of Available Vessels (K)	$\in [3, 20]$
	Demand to Capacity Ratio (DCR)	$\in \{85, 95, 105\}$
	Instance Number (I)	$\in [1, 5]$

Table 5.5: In order to clearly differentiate between instances a instance reference scheme were created, naming instances according to the seven dimensions that varies.

5.4 Test instances

The generator outlined above has several random components. To ensure consistency and comparability, we create a finite number of test instances. To mitigate the effects randomness can have on test results, we generate a minimum of five instance for each combination of dimension values. Table 5.6 shows an overview of all test instance groups used throughout this thesis, with their reference name and all of their dimension values.

All test instances has a planning horizon length of four weeks, independent of how many voyages are being tested. The reason for this is that our case company usually have even larger number of voyages than we use for planning horizon's of the same length.

#	Instance group name	Port	Contract	Voyage	Vessel	DpC	Count
1	AmAs_5_C_4_4_95	5	$C \in$ { 10, 20, 30, 50, 100, 150 }	4	4	95%	30
1	AsE_10_C_3_3_95	10	$C \in$ { 10, 20, 30, 50, 100, 150 }	3	3	95%	30
1	AsE_10_C_4_4_95	10	$C \in$ { 10, 20, 30, 50, 100, 150 }	4	4	95%	30
1	AsE_10_C_5_5_95	10	$C \in$ { 10, 20, 30, 50, 100, 150 }	5	5	95%	30
1	AsE_10_C_6_6_95	10	$C \in$ { 10, 20, 30, 50, 100, 150 }	6	6	95%	30
1	EAm_15_C_4_4_95	15	$C \in$ { 10, 20, 30, 50, 100, 150 }	4	4	95%	30
1	AsE_10_100_3_3_DcR	10	100	3	3	$DcR \in$ [94%,126%]	140
1	AsE_10_C_3_3_DcR	10	$C \in$ [100, 180]	3	3	dependent on C	205

Table 5.6: The different groups of test instances that are being used throughout the computational study.

AsE_10_C_3_3_95, AsE_10_C_4_4_95, AsE_10_C_5_5_95, AsE_10_C_6_6_95,with $C \in \{10, 20, 30, 50, 100, 150\}$.

5.5 Summary

In Section 5.1 to 5.2 we outline how instances are generated using input data and randomly generated values. We show how the generator creates a fleet using real data, which is then used by the generator to sample vessel for instances. In addition, we described how our networks are generated with real data to simulate real trade routes, and how our contract generator uses domain knowledge to create realistic contracts. In order to clearly differentiate between instances, a reference scheme was developed with reference names based on the seven dimensions that can vary from instance to instance. The reference scheme is outlined in Section 5.3, while a list of all test instance groups used in this thesis can be seen in Section 5.4.

Chapter 6

Computational Study

This Chapter contains results from the performance analysis of the model presented in Chapter 4. The goal of this analysis is to examine if the model can be solved in reasonable time when model instances are scaled up to the size of realistic problem cases. In the first section, 6.1 we evaluate how the model performance is affected by including transit time and evenly spread restrictions. In Section 6.2, we examine the model performance in terms of run time when systematically varying the number of ports and the number of contracts. In the next Section, 6.3, we will evaluate the model performance when systematically varying the number of maximum voyages and the number of contracts. In Section 6.5, we examine the model performance is affected when varying penalty unit cost of deviating from evenly spread restrictions. In the last Section, 6.6, we present our conclusion of the performance analysis.

All test instances used in the mathematical programming model have been generated using a Python version 3.5.2 script, which is detailed in Section 5. All test instance are solved using a Java 1.8 program which uses the Mosel Xpress API, or with the Mosel Xpress program. Each model run is terminated after 1800 seconds unless specified otherwise. The model has been run on a Windows 10 computer with an Intel i5 Quad Core, 3.3 GHz processor, 8 GB DDR3 1600 MHz PC3-12800 RAM, and an S-ATA 6Gb/s SSD disk.

6.1 Transit time and evenly spread requirements effect on run time

In this section, we examine how performance in terms of run time is affected when we add transit time and evenly spread constraints to the model. We run relaxed versions of the complete model with a set of test instances, and compare the results with runs of the complete model with the same instances. The instances used for this analysis are generated from three different trade routes, the America to Asia trade route with five ports, the Asia to Europe trade route with ten ports, and the Europe to America trade route with 15 ports. Each of the instances generated take on a contract number value from the set $\{10, 20, 30, 50\}$. We generated five instances for each combination of ports number and contracts number. The specific instances used are AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 where $C \in \{10, 20, 30, 50\}$. Results from running the model without transit time and evenly spread constraints are aggregated and summarized in Table 6.1 and Table 6.2 respectively; the results for running the complete model are aggregated and summarized in Table 6.3, for comparison. All tables show the number of ports, the number of contracts, average run time, average optimality gap and the number of instances which solves to optimality within the run time limit. The run time limit for this analysis is 1800 seconds.

	5 ports			**10 ports**		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	2.6 s	0.0%	5/5	4.4 s	0.0%	5/5
20	1.4 s	0.0%	5/5	3.0 s	0.0%	5/5
30	10.0 s	0.0%	5/5	8.2 s	0.0%	5/5
50	16.0 s	0.0%	5/5	15.5 s	0.0%	5/5

	15 ports		
Cont	**Run time**	**Gap**	**Optimal**
10	2.6 s	0.0%	5/5
20	0.8 s	0.0%	5/5
30	9.4 s	0.0%	5/5
50	30.4 s	0.0%	5/5

Table 6.1: Results from running instance groups AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 where $C \in \{10, 20, 30, 50, 100, 150\}$, without transit time constraints.

From Table 6.1 we can see that removing transit time constraints drastically reduces average run time compared to the results in 6.3, indicating that the transit time constraints adds significant complexity to our model.

From Table 6.2 we can see that removing evenly spread constraints significantly increases average run time compared to the results in 6.3, in contrast to the effect of removing transit time constraints. Our hypothesis for explaining these results is that the evenly spread constraints limit the search space for the t-variable, as well as the

	5 ports			10 ports		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	9.2 s	0.0%	5/5	278.4 s	0.0%	5/5
20	36.0 s	0.0%	5/5	583.2 s	0.0%	5/5
30	37.6 s	0.0%	5/5	867.6 s	0.0%	5/5
50	136.4 s	0.0%	5/5	1049.8 s	0.0%	5/5

	15 ports		
Cont	**Run time**	**Gap**	**Optimal**
10	53.8 s	0.0%	5/5
20	>1505.2 s	2.9%	5/5
30	>1284.4 s	2.5%	5/5
50	>1549.2 s	2.5%	5/5

Table 6.2: Results from running instance groups AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 where $C \in \{10, 20, 30, 50, 100, 150\}$, without evenly spread constraints.

	5 ports			10 ports		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	4.4 s	0.0%	5/5	12.4 s	0.0%	5/5
20	12.6 s	0.0%	5/5	26.8 s	0.0%	5/5
30	21.2 s	0.0%	5/5	53.0 s	0.0%	5/5
50	481.8 s	0.0%	5/5	78.6 s	0.0%	5/5

	15 ports		
Cont	**Run time**	**Gap**	**Optimal**
10	12.6 s	0.0%	5/5
20	29.2 s	0.0%	5/5
30	41.2 s	0.0%	5/5
50	345.2 s	0.0%	5/5

Table 6.3: Results from running instances AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 where $C \in \{10, 20, 30, 50, 100, 150\}$.

delta-variable and other voyage dependent variables indirectly. Consider a contract for a given port that requires more than one pickup. When the model selects a pickup for that contract for a given voyage, it indirectly and strictly limits which other voyages and at which times the other pickups must happen. In our model pickups for contracts with evenly spread requirements are restricted to happen after a certain time period from the last pickup plus an acceptable slack, which in our case is plus or minus three days. The limiting effect grows for each contract pickup that is selected, continuously and increasingly reducing the search space for each pickup.

6.2 Varying the number of contracts and ports

In this section, we examine how performance in terms of run time is affected by systematically increasing the number of contracts and the number of ports. The instances used for this analysis are generated from three different trade routes, the America to Asia trade route with five ports, the Asia to Europe trade route with ten ports, and the Europe to America trade route with 15 ports. Each of the instances generated take on a contract number value from the set $\{10, 20, 30, 50, 100, 150\}$. We generated five instances for each combination of ports number and contracts number. The specific instances used are AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 with $C \in \{10, 20, 30, 50, 100, 150\}$. Results are aggregated and summarized in Table 6.4, which shows the number of ports, the number of contracts, average run time, average optimality gap and the number of instances which solves to optimality within the run time limit. The run time limit for this analysis is 1800 seconds.

The results in 6.4 are slightly distorted by outlier instances, which are easier or more difficult to solve than the average instance in its group. For example, the average run time for the instances with five ports and 50 contracts is much larger than what would be expected when considering the entire table and is the result of one extreme outlier instance. The same is true for the instances with 15 ports and 50 contracts, which also has one extreme outlier instance. If we remove both these outlier instances, the adjust average run time for the instances with five ports and 50 contracts are 53.0 seconds, while it is 154.8 for the instances with 15 ports and 50 contracts. These results seem more reasonable in the context of the entire table. If we ran a much larger set of test instances, the effect of outliers would be averaged out.

From the results in 6.4 we can see that increasing the number of contracts will increase the run time non linearly. This is true for all possible port number values. The relationship between the number of ports and run time is not as clear. For a number of contracts between 10 and 30, there is a clear increase between five ports and 10

	5 ports			10 ports		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	4.4 s	0.0%	5/5	12.4 s	0.0%	5/5
20	12.6 s	0.0%	5/5	26.8 s	0.0%	5/5
30	21.2 s	0.0%	5/5	53.0 s	0.0%	5/5
50	481.8 s	0.0%	5/5	78.6 s	0.0%	5/5
100	>665.8 s	0.01%	4/5	>531.8 s	0.2%	4/5
150	>1799.2 s	2.0%	0/5	125.2 s	0.0%	5/5

	15 ports		
Cont	**Run time**	**Gap**	**Optimal**
10	12.6 s	0.0%	5/5
20	29.2 s	0.0%	5/5
30	41.2 s	0.0%	5/5
50	345.2 s	0.0%	5/5
100	>483.6 s	0.2%	4/5
150	>868.0 s	0.02%	3/5

Table 6.4: Results from running instances AmAs_5_C_4_4_95, AsE_10_C_4_4_95, EAm_15_C_4_4_95 where $C \in \{10, 20, 30, 50, 100, 150\}$.

ports, at least if you adjust for outliers. However, the same increase can not be seen between 10 ports and 15 ports. In fact, the change in run time does not seem to be statistically significant. This is true for 50 contracts as well, when you adjust for outliers. For a number of contracts above 50, this relationship becomes less clear. The instances with five ports solve to optimality less often and have larger optimality gaps on average than instances with 15 ports. From the presented results, one might expect that instances with fewer ports should be less complex and solve faster. In our analysis, instances with five ports are compared to instances with 10 and 15 ports using the same number of contracts. That is, run time for instances with five ports and 30 contracts are compared with run time for instances with 15 ports and 30 contracts. Our hypothesize to explain these results is that when the number of contracts per port increases, the difficulty of finding a solution satisfying contract evenly spread requirements increases as well. When there are more contracts per port, there are more evenly spread requirements per port, and more possible ways to choose a set of contracts to deviate from when the requirements are non compatible. This creates a larger search space when the model has to select a port visit time to limit the deviation from evenly spread requirements.

6.3 Varying the number of contracts and maximum voyages

In this section, we examine how performance in terms of run time is affected by systematically increasing the number of contracts and the maximum number of voyages. The instances used for this analysis are generated from the Asia to Europe trade route with ten ports. Each of the instances generated take on a contract number value from the set $\{10, 20, 30, 50, 100, 150\}$, and a maximum number of voyages value from the set $\{3, 4, 5, 6\}$. We generated five instances for each combination of maximum number of voyages and contracts number. The reference names for the instances used are AsE_10_C_3_3_95, AsE_10_C_4_4_95, AsE_10_C_5_5_95, AsE_10_C_6_6_95,with $C \in \{10, 20, 30, 50, 100, 150\}$. Results are aggregated and summarized in Table 6.5, which shows the number of ports, the number of contracts, average run time, average optimality gap and the number of instances which solves to optimality within the run time limit. The run time limit for this analysis is 1800 seconds.

	3 voyages			**4 voyages**		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	3.2 s	0.0%	5/5	12.4 s	0.0%	5/5
20	2.4 s	0.0%	5/5	26.8 s	0.0%	5/5
30	7.2 s	0.0%	5/5	53.0 s	0.0%	5/5
50	15.8 s	0.0%	5/5	78.6	0.0%	5/5
100	37.4 s	0.0%	5/5	531.8 s	0.2%	4/5
150	51.8 s	0.0%	5/5	125.2	0.0%	5/5

	5 voyages			**6 voyages**		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	>729.4 s	0.1%	3/5	>773.6 s	0.3%	4/5
20	300.8 s	0.0%	5/5	>1799.4 s	0.5%	0/5
30	>645.2 s	0.4%	4/5	>1799.4 s	0.5%	0/5
100	567.4 s	0.0%	5/5	>1799.6 s	0.9%	0/5
150	1465.6 s	3.7%	2/5	>1799.4 s	14.3%	0/5

Table 6.5: Results from running instances AsE_10_C_3_3_95, AsE_10_C_4_4_95, AsE_10_C_5_5_95, AsE_10_C_6_6_95,where $C \in \{10, 20, 30, 50, 100, 150\}$

From the results in 6.5 we can see that the average run time value for instances with 10 contracts and five voyages are are larger than you would expect when considering the entire table, and is caused by one extreme outlier. If we remove the outlier instance, the adjusted average run time for 10 contracts and five voyages becomes 81.0 seconds, which seems more reasonable in when compared with of the rest of the results. In addition, we would expect the average run time for 100 contracts and four voyages to be higher, which again shows how large differences can be in run times for instances with the same configuration.

However, if we take the entire table into consideration there is a clear trend. We can see that increasing the maximum number of voyages increases the run time non linearly. Observe that increasing the number of contracts for instances with a given maximum voyage number will also increase the run time. Instances with three, or four voyages as maximum voyage number are all solved to optimality within the run time limit, with only one exception for one of the instances with 100 contracts and four as maximum voyage number. For instances with five as maximum voyage number, six of 25 cannot be solved to optimality within the run time limit. Of instances with six as a maximum voyage number, only four instances of 25 solve to optimality and they all belong to the instance group with 10 contracts. For those instances which cannot be solved, the average optimality gap increases as the number of contracts increases.

6.4 Varying the number of contracts and vessels

In this section, we examine how performance in terms of run time is affected by systematically increasing the number of contracts and the number of available vessels. The instances used for this analysis are generated from the Asia to Europe trade route with ten ports. Each of the instances generated take on a contract number value from the set $\{10, 20, 30, 50\}$, and a number of available vessels value from the set $\{8, 12, 16\}$. We generated five instances for each combination available vessels number and number of contracts number. The reference names for the instances used are AsE_10_C_4_8_95, AsE_10_C_4_12_95, AsE_10_C_4_16_95, with $C \in \{10, 20, 30, 50\}$. Results are aggregated and summarized in Table 6.6, which shows the number of ports, the number of contracts, average run time, average optimality gap and the number of instances which solves to optimality within the run time limit. The run time limit for this analysis is 1800 seconds.

From the results in 6.6 we can see that increasing the number of available vessels increases the run time non linearly, and run time increases as a function of the number of contracts as well. The model is able to solve instances up to 12 available vessels within run time, with exception of those instances that have contract number of 50. Instances with 16 available vessels are less likely to be solved within the run time limit, and the optimality gap for those that are not solved increases as a function of the number of contracts.

	8 vessels			12 vessels		
Cont	**Run time**	**Gap**	**Optimal**	**Run time**	**Gap**	**Optimal**
10	15.0 s	0.0%	5/5	67.6 s	0.0%	5/5
20	139.6 s	0.0%	5/5	140.2 s	0.0%	5/5
30	146.4 s	0.0%	5/5	480.0 s	0.0%	5/5
50	298.2 s	0.0%	5/5	1765.6 s	1.3%	1/5
100	1358.2 s	3.9%	3/5	1799.4 s	6.2%	0/5
150	1755.0 s	2.1%	1/5	1799.4 s	9.6%	0/5

	16 vessels		
Cont	**Run time**	**Gap**	**Optimal**
10	>743.6 s	0.1%	3/5
20	>631.0 s	0.1%	4/5
30	>937.2 s	0.2%	4/5
50	>1799.4 s	1.3%	0/5
100	>1800.0 s	13.1%	0/5
150	>1799.6 s	11.3%	0/5

Table 6.6: Results from running instances AsE_10_C_4_8_95, AsE_10_C_4_12_95, AsE_10_C_4_16_95, with $C \in \{10, 20, 30, 50, 100, 150\}$.

6.5 Varying evenly spread penalty cost

In this section, we evaluate how performance in terms of run time is affected by varying penalty costs on deviation from the evenly spread restrictions. Figure 6.1 shows the results of an example instance that is run with different levels of unit penalty cost (UPC), represented with a histogram. The horizontal axis shows UPC values. The left vertical axis shows run time in percentage increase from the lowest run time, while the right axis and the red graph line, which is superimposed over the histogram, shows increases in operational costs as a function of UPC.

The results seem to show a pattern of variation in run time as UPC is increased. Run times are lower when there are no penalty costs and operational costs dominate. As UPC is increased and the model has to start making trade offs between adhering to evenly spread restrictions and minimizing operational costs, run time increases. When penalty costs increase to a point where it dominates operational costs by a large factor, limiting the search space, the run times decreases again. Observe that this decrease coincides with the point where operational costs reach its highest level and stabilize. From this point on, penalty costs start to dominate operational costs, and there is little to no change in the solution. In Figure 6.2 we can see the average run

Figure 6.1: Instance AsE_10_100_4_4_95_3, run with different levels of unit penalty cost. The horizontal axis shows unit penalty cost values. The left vertical axis shows run time in percentage increase from the lowest run time, while right axis and the red graph line shows increases in operational costs as a function of unit penalty costs.

time for four instances from the group AmAs_5_100_4_4_0.95 as a function of UPC. Observe that the trend in run time is the same here, as in the example.

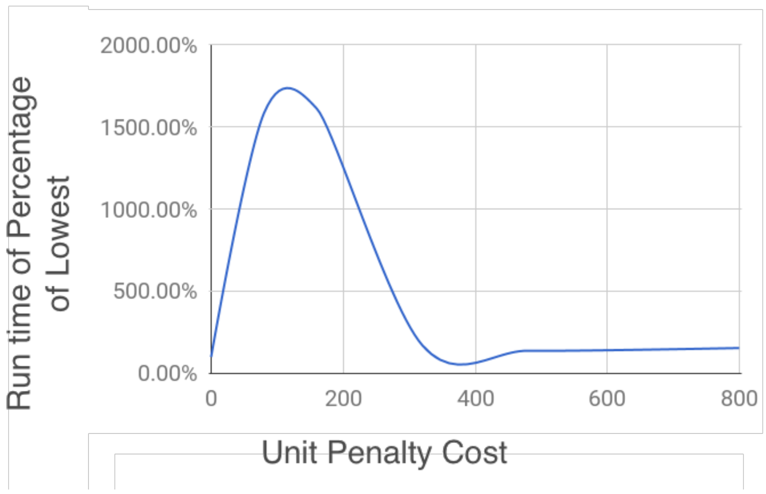


Figure 6.2: the average run time for four instances from the group AmAs_5_100_4_4_-0.95 as a function of UPC.

6.6 Summary

In this chapter, we have analyzed how performance in terms of run time is affected by varying the five instance dimensions: number of ports, number of contracts, number of maximum voyages, number of available vessels, and unit penalty costs. The results show that run time increase non linearly for all dimensions, and that solving instances larger than relatively small test instances used in this thesis within a reasonable time limit is not possible.

Chapter 7

Example solution analysis

In this chapter we will analyze one instance solution in depth, to see if the results are as we would expect from our model formulation and the solutions are reasonable when related to the real problem. The specific instance used is AsE_10_150_4_4_95_1, which we ran both with low and high penalty costs for deviating from evenly spread restrictions. In the first section, Section 7.1, we will examine the behavior of the evenly spread constraints and introduce a diagram representation for visualizing the logistics of the resulting planning horizon, including which ports are visited, at what times they are visited, and which speeds were used by the selected vessels during its corresponding voyage. Section 7.2 contains an analysis of the behavior of the transit time constraint. In Section 7.3 we will study the behavior of the loading constraints and introduce a heat map representation visualizing the use of available stowage capacity for the entire planning horizon. Both representations will be used through our analysis chapters to visualize results from model run.

7.1 Evenly spread constraints analysis

In this section, we will examine the behavior of the evenly spread constraints both with low and high deviation penalty cost. Each day of deviation get multiplied by a unit penalty cost (UPC), and the result adds to the total deviation penalty costs. Hence, by varying the unit penalty cost (UPC), we affect how the model prioritizes its two objectives: minimizing operational costs and spreading contract pickups evenly. Figures 7.1 and 7.2 both shows a diagram representation for visualizing the logistics of the resulting planning horizon, including which ports are visited, at what times they are visited, and which speeds were used by the selected vessels during its corresponding voyage. The figures represent the solution for the same instance with low and high

UPC, respectively. Figure 7.1 includes a map representation of the trade route used for this instance, the Asia to Europe trade route.

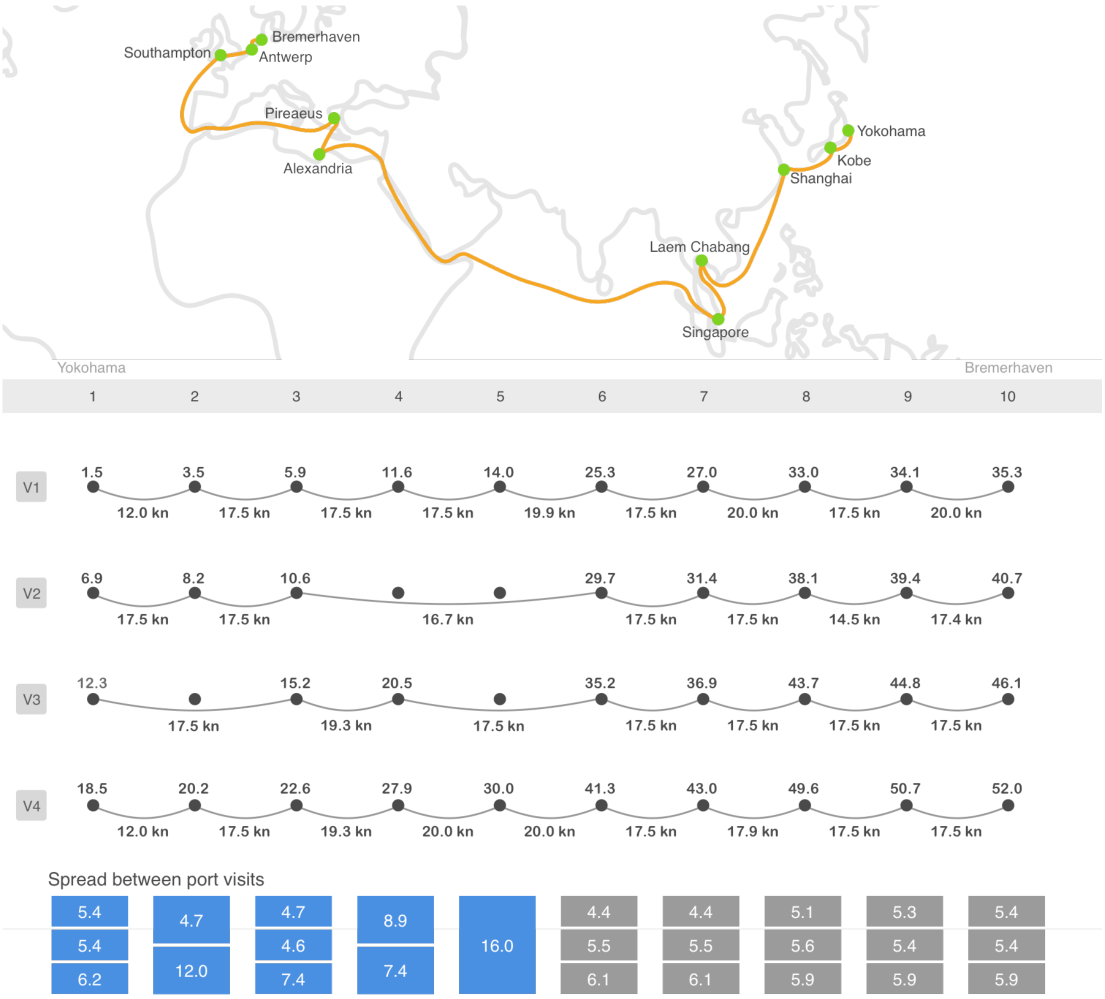


Figure 7.1: Diagram representation for instance AsE_10_150_4_4_95_1 with low UPC.

Each Figure has a row of nodes and arcs, each row corresponding to a voyage. The nodes correspond to ports, and the arcs are between those ports that are visited consecutively. Above each node is a label which shows what time the port was visited. Each arc has a label, which shows at what speed the arc was sailed. At the bottom of both figures, there is a footer which shows the time difference between port visits to the same port for different voyages.

Fairly evenly spread restrictions are for contracts, not for port visits, and the diagrams do not include direct information about how well the contracts are spread. However, we can observe the restrictions in action as they indirectly affect port visits as well.

For example, let us examine the time differences between the four port visits to port one in Figure 7.2. Contracts with a pickup frequency of four should in accordance with our model have a spread of $^{28}/_{4} = 7 \pm 2$ days. We can see from the figure that all port visits for the same port on different voyages have a time difference between visits which conforms to this requirement as well. Similar observations can be made for the other ports, with only a few days of deviation in total from the evenly spread requirements. The deep sea leg of the trade route used for this instance is between port five and six, which means that only contracts from the first five ports are tonnage driving contracts, meaning that only those contracts have evenly spread restrictions. However, we can observe from the figure that all port visits are generally fairly evenly spread in time. The explanation for this is that all contracts have transit time restrictions which force the goods that are picked up per the requirements of contracts to be unloaded at the destination port of the goods within a limited amount of time after being loaded in their origin port. These constraints in combination with the evenly spread constraints force an approximate evenly spreading of port visits over the entire planning horizon.

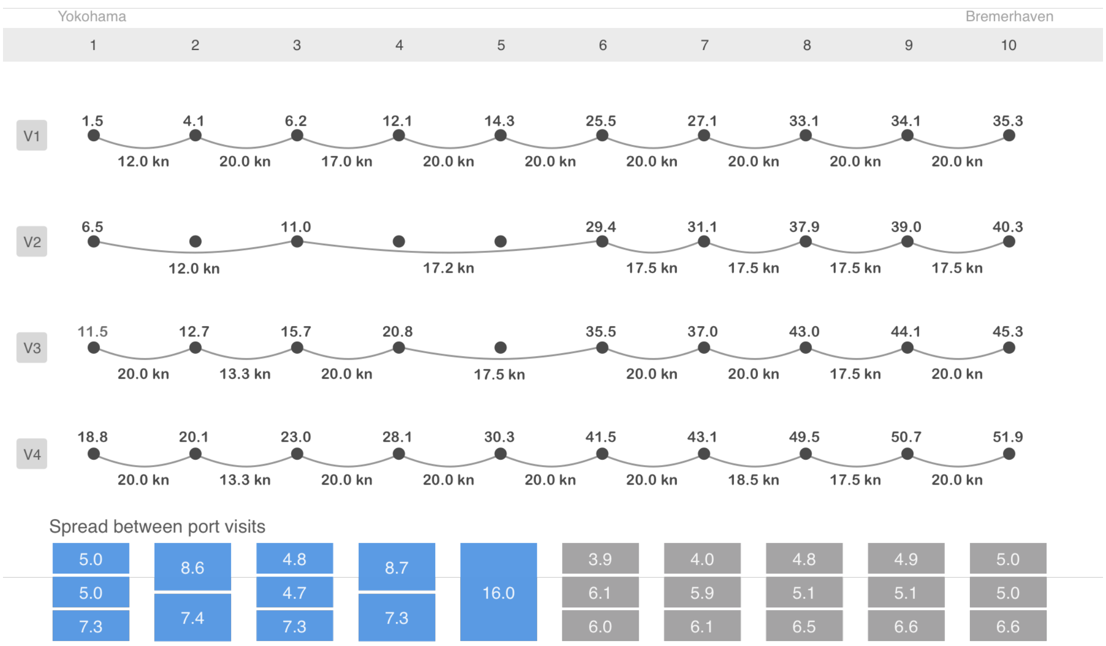


Figure 7.2: Diagram representation for instance AsE_10_150_4_4_95_40_1 with high UPC.

To get a sense of how well contracts for a given solution are evenly spread, we define a measure called deviation per contract. The evenly spread constraints may be deviated from by incurring penalty costs. Deviation per contract (DpC) is the sum of deviation time for each contract divided over the number of contracts. As an example of usage, when we run the instance with a UPC of 0.8, the DpC is 0.98, while it is 0.92 when run with a UPC of 40. For comparison, the DpC is 1.49 when the instance is run with no penalty costs. All though it is an aggregated measure, it gives us an approximation of the effect on evenly spreading when increasing the UPC. Table 7.1 contains all contracts with evenly spread deviation for our example instance run with high UPC. We can see from the table that 15% of all contracts deviate from evenly spread constraints; 7% of all contracts deviate from the constraints by less than one day, while 8% of all contracts severely deviates from the constraints by roughly one week. These results are a consequence of assigning the contracts with penalty costs which are proportional to evenly spread priority: to lower total deviation penalty costs the model chooses to severely deviate from evenly spread constraints for the contracts with lower penalty costs. A possible way to avoid this behavior would be to have the slack cost be dependent on how many days the contract deviates, as well as priority.

Contract number	**Deviation time (days)**	**Penalty cost**
2	6.7	106%
7	0.2	247%
14	7.7	106%
21	0.22	217%
24	0.22	196%
29	6.7	103%
33	6.7	104%
39	0.22	204%
41	7.7	109%
47	6.7	106%
56	7.2	100%
63	0.88	149%
70	0.88	129%
84	7.7	102%
86	0.22	195%

Table 7.1: All contracts that deviates from evenly spread restrictions, with the number of days they deviate from being evenly spread, and the corresponding slack cost per day in percentage of lowest. Observe that the contracts with lower slack costs deviates more.

Improving the DcP does come at a cost: compared to running the instance with no penalty cost, running the instance with high UPC increases the operational costs with 1.88%. There are two factors that affect operational costs in our model: port visit costs which depend on the number of port visits, and sailing costs which depend on sailing speed. We can verify that sailing costs are the cause of cost increase in this case by using the figures. From Figures 7.1 and 7.2 we can see that both instances visit each port exactly the same number of times, and that the sailing speeds over arcs generally increases rather than decreases. In fact, the average sailing speed increases from 18.34 knots to 18.77 knots for low and high cost, respectively. This offers us some insight into one of the mechanisms the model uses to minimizing evenly spread deviation when the model is required to prioritize this objective sufficiently above operational costs. That is, tactically varying sailing speeds above what would be cost efficient with no evenly spread constraints to avoid costly deviations. Other mechanisms to lower evenly spread deviations is to reschedule contract pickups, which can be observed in Figure 7.5 and 7.6 in Section 7.3, as well as adding port visits to the planning horizon.

The fact that operational costs increases with higher UPC suggest an interesting economic use case: what is the relationship between operational costs, DpC and UPC. In Section 8.1 we will explore this use case further, and generate trade off curves for the two objectives of minimizing operational costs and improving evenly spreading.

7.2 Transit time constraints analysis

In this section, we will examine the behavior of the transit time constraints, with both low and high unit penalty cost (UPC). Each contract has a corresponding transit time constraint which forces the goods that are picked up per the requirements of contracts to be unloaded at the destination port of the goods within a limited amount of time after being loaded in their origin port. For goods that are unloaded in their destination port before they are required to be unloaded, we will use the term transit time slack to mean the remaining period of the transit time requirement. For example, goods loaded in connection with a contract A is loaded at day four and unloaded at day ten. Contract A has a transit time limit of ten days. In this case, the transit time slack would be four days.

Figures 7.3 and 7.4 both visualizes the transit time slack distribution by using a histogram. Since transit time slack values are continuous, we have divided the entire range of transit time slack into a series of intervals, called bins, with the length of one day, each starting at an integer value. The horizontal axis shows bins, while the vertical axis shows how many contract pickups have transit time slack which falls within the interval of each bin. As an example, a contract pickup with transit time slack of 4.4 belongs to the 4-5 bin. This increases the height of the bar belonging to that bin with one.

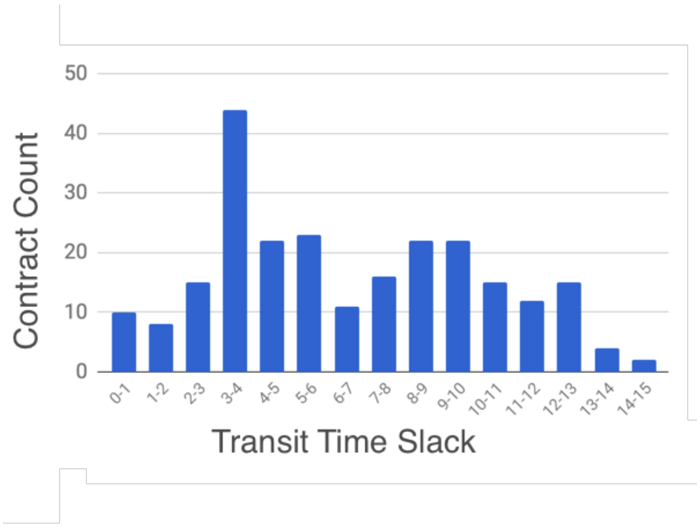


Figure 7.3: Distribution of transit time slack for instance AsE_10_150_4_4_95_1 with low penalty cost.

We can see from the histograms that the transit time slack for all contract pickups are positive, indicating that the transit time restrictions are functioning as expected. Another thing to observe is that roughly 86% of contracts have three or more days of slack, which leads to the question: why doesn't the model exploit all the available transit time slack and decrease speeds to improve operational costs. The answer is that the contract pickups that has strict transit times indirectly enforce strict transit times on the other contract pickups as well. As an example, if during a voyage the contract with the strictest transit time requirements loads at port one and unloads at the last port, the entire voyage affected. Even if this is strict transit time requirement pertains to a contract with only one pickup, the evenly spread requirements of other contracts with more pickups transfers this strict transit time requirement over all voyages. Together these constraints limit the model's ability to schedule contract pickups to exploit the transit time slack available.

If we compare Figure 7.3 and Figure 7.4, we can see a slight difference when UPC increases from low to high, respectively: a reduction in the first four bins, except bin 1-2 which increase by one, while the rest of the bins either increases or remains the same. The results are consistent with previous results: as was shown in Section **??**, high UPC causes the model to increase speeds to avoid deviating from evenly spread restrictions. Increases in speed mean earlier arrival for some port visits, and therefore also larger transit time slack for some contract pickups.s.

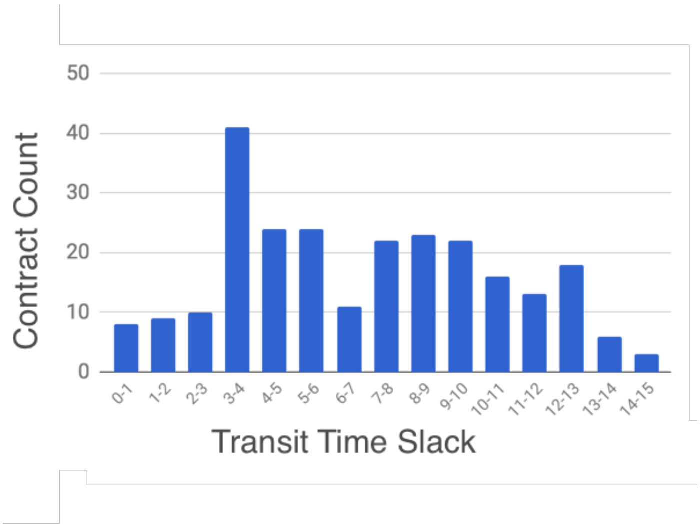


Figure 7.4: Distribution of transit time slack for instance AsE_10_150_4_4_95_1 with high penalty cost.

7.3 Capacity constraints analysis

In this section, we will examine the behavior of the loading and capacity constraints, with low and high unit penalty costs (UPC). Figures 7.1 and 7.1 visualizes the use of available stowage capacity for the entire planning horizon using heat map representations for low and high UPC, respectively. Each voyage of the planning horizon has its own heat map. All cells in one row show loaded quantity of the same product type, while all cells in the same column show loaded quantity when the vessel for that voyage travel between two ports. Note that this does not necessarily mean the vessel visit those two ports. Hence, each cell in the heat map shows used capacity for the product its row represents, between the two ports its column represents.

Further more, observe from Figure 7.5 that the vessels are at peak capacity only during a limited part of the voyage, which for most voyages is near full or full capacity. All voyages have the highest level of used capacity over the deep sea leg: the vessels for voyage one and four are at peak capacity between ports four and seven, while the vessel for voyage four is at peak capacity between ports four and six. This seems reasonable, considering that for our instances at least 70% of an instance's contracts are tonnage driving contracts, meaning that they are loaded before the deep sea leg, and unloaded at some port the deep sea leg. This means that there is underutilized capacity for parts of each voyage, which might be used to pick up more wayward and spot contracts. Also, observe that voyage three has very low capacity utilization

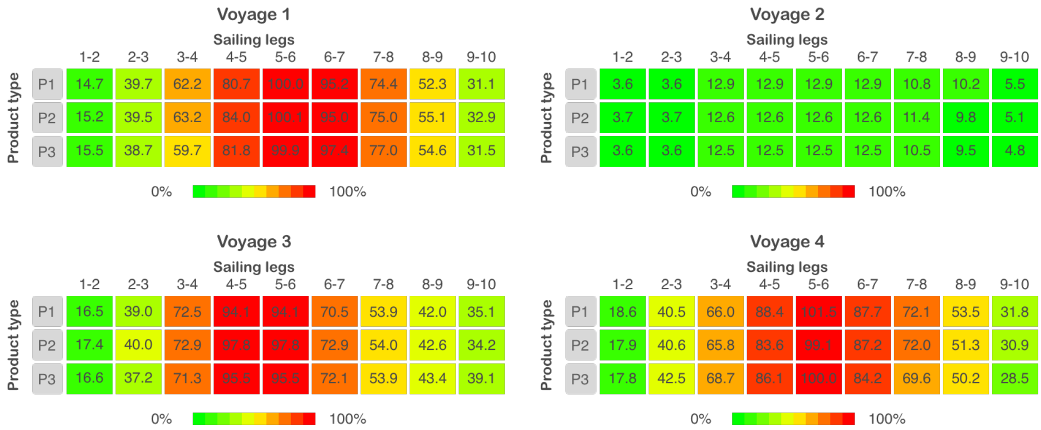


Figure 7.5: Heat map representation for instance AsE_10_150_4_4_95_1 with low penalty cost.

during its deep sea leg. The explanation for this is that not all contracts are loaded and stowed at the exact time; unloading goods in a port frees up space for loading new goods in the same port and every port after that. In other words, the vessel capacity is a measure of the quantity that can be stowed on the vessel at a specific time during a voyage, not a measure of the total quantity of goods that can be transported in total over the entire voyage. There are two reasons for a voyage with low capacity utilization to be scheduled at all by the model: either the vessels available does not have enough capacity to pick up all demand in less than the assigned number of voyages, or one or more contracts requires a pickup for every possible voyage during the planning horizon. In this specific instance, we know that there are 14 contracts that require a pick up for each of the four voyages, possibly forcing the model to schedule a low capacity voyage.

These two observations suggest two interesting economic use cases: First, examine how feasibility and cost per unit of goods are affected by increasing the quantity of goods transported. Second, examine if costs can be reduced by removing low capacity voyages while keeping demand equal. In Section 8.2 and Section 8.3, respectively, we will explore both cases.

We can also observe the effects of the evenly spread requirements on loading patterns. For example, notice the difference in used capacity in Figure 7.5 and 7.6 in voyage one and four, between port five and six. The explanation for these changes is that some subset of contracts has been rescheduled to limit penalty costs for spread deviation.

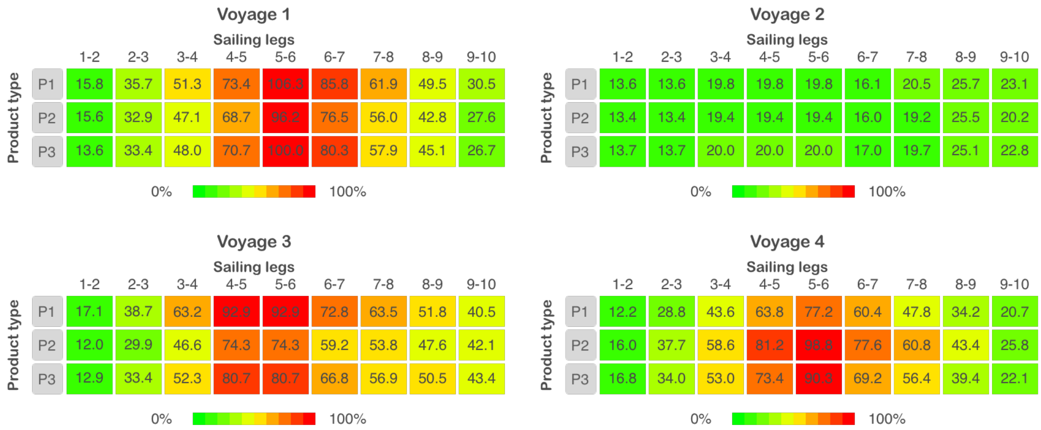


Figure 7.6: Heat map representation for instance AsE_10_150_4_4_95_1 with high penalty cost.

7.4 Summary

In summary, we have seen that the evenly spread, transit time and loading constraints all work as expected. The evenly spread constraints managed to spread most of the contracts with little or no deviation, as well as indirectly and approximately spreading port visits as well. However, the contracts that had lowest evenly spread prioritization tended to deviate severely, as much as seven to eight days in most cases. When the model was run with a high prioritization of evenly spread requirements, operational costs increased with 1.88% as a result of increased sailing cost. Deviation, measured in days of deviation per contract, decreased from 1.49 with no prioritization of spread requirements, to 0.92. We saw that a large percentage of transit time constraints had abundant of transit time slack, and only a few transit time constraints were close to breaking their corresponding contracts' requirements. However, these few constraints, in combination with evenly spread constraints for the corresponding contracts, indirectly enforced strict transit times on the other contract pickups. We also learned that efficient scheduling of pickups led to one out of four voyages having low capacity utilization, while the rest had near of full capacity utilization at peak capacity. The voyage with low capacity utilization had roughly 13% capacity utilization at peak, for an instance with a total demand quantity that was equal to 95% of available stowage capacity.

Based on this analysis, three interesting economic cases were suggested. First, what is the relationship between operational costs, deviation per contract and different levels of prioritization of evenly spread. Second, examine how feasibility and cost per unit of good are affected by increasing the quantity of goods transported. Third, examine if costs can be reduced by removing low capacity voyages while keeping demand equal.

Chapter 8

Practical Use and Managerial Insights

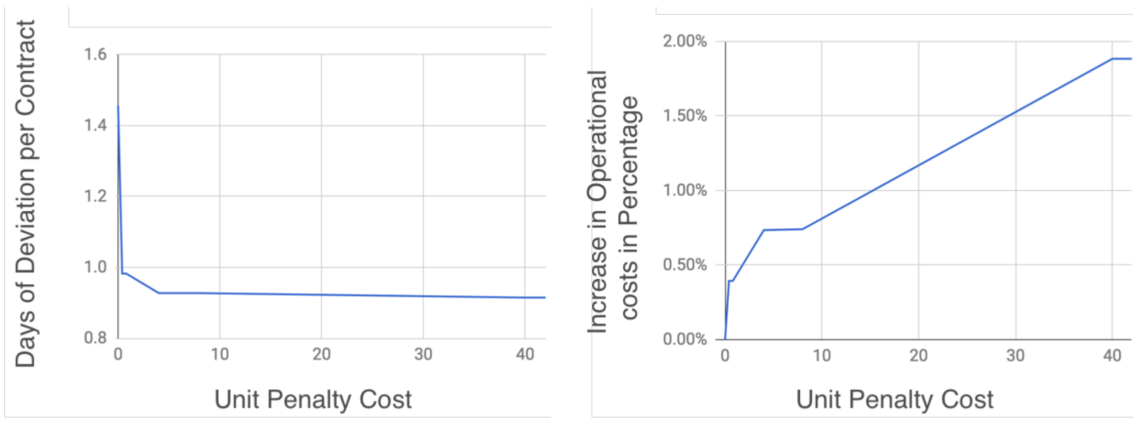
In this chapter, we will analyze model solutions for selected instances, with the goal of exploring practical applications of the model and gaining managerial insights which can be applied to real life situations.

In the four first sections we will look at the three economic use cases we proposed in Section 7, as well as a fourth economic use case suggested by the case company. In the first section, Section 8.1, we examine how operational costs are affected by varying how the model prioritizes its two separate objectives: evenly spreading contract pickups and minimizing operational costs. We do this by systematically changing the unit penalty cost, which increases the cost of deviating from evenly spread requirements. In Section 8.2 we systematically increase total demand to explore how sensitive the model is to an increasing demand to capacity ratio (DCR), as well as considering the relationship between the DCR and cost per unit of a good. In the next section, 8.3, we examine how removing low capacity voyages from a planning horizon affects operational costs. In section, 8.4, we consider a case suggested to us by the case company, and explore if costs can be reduced by merging two planning horizons over the same trade route into one single planning horizon. In Section 8.5 In we will examine if operational costs can be reduced by making transit time requirements less strict. In the last section, 8.6, the findings of this chapter are summarized.

8.1 Varying penalty cost of deviating from evenly spread restrictions

In this section, we examine how operational costs are affected by varying how the model prioritizes its two separate objectives: evenly spreading contract pickups and minimizing operational costs. We do this by systematically changing the unit penalty cost, which increases the cost of deviating from evenly spread requirements. We show how the model can be used to approximate a Pareto front, which allows decision makers to chose a preferred trade-off between the two objectives.

To make an appropriate trade-off decision we have to consider two things: improvement in DcP and increase in operational costs. In Section 7.1 we studied an example instance and examined the difference between solutions for two levels of UPC. Based on those results we would expect DcP to decrease and operational costs to increase.



(a) (b)

Figure 8.1: (a) Demand to capacity ratio as a function of unit penalty cost (UPC) over the interval [0,40] of the UPC domain. (b) increase in operational costs, as a percentage of the cost with no UPC, and as a function of UPC over the interval [0,40] of the UPC domain.

In Figure 8.1 (a) we present the DcP as a function of UPC over the interval [0,40] of the UPC domain. By increasing UPC we have an immediate sharp decrease in DcP between zero and four, and after that, there is an almost negligible improvement in DcP between four and 40. Beyond 40 DcP stabilizes and shows no improvement as UPC continues to increase. The reason for this is that the evenly spread requirements for the contracts in this instance are incompatible, and there are no possible optimal solutions that fulfill every evenly spread requirement. The model has to deviate from evenly spread requirements for a subset of contracts and selects this subset by finding an optimal balance between minimizing the number of contracts that has its requirements broken and prioritizing to satisfy evenly spread requirements for contracts with higher penalty costs.

Improving deviation per contract should be considered in relation to the increasing cost of that improvement. In Figure 8.1 (b) we present the increase in operational costs, as a percentage of the cost with no UPC, and as a function of UPC over the interval [0,40] of the UPC domain. By increasing UPC from zero to four, we can see an increase in operational costs of 0.75%, which remains stable between four and eight. From eight to 40 we can see a continues increase from 0.75% to 1.88%. Beyond 40 it stabilizes on a 1.88% increase in operational costs.

Observe that improvements in DcP are small when increasing UPC above four while operational costs continue to increase from eight to forty. This might make a UPC value between four and eight a natural choice of value for this instance. All though the increase in operational cost from achieving the best possible DcP is only 1.88%, this will still be a large increase in terms of monetary value as the magnitude of the costs are quite large. However, depending on how much the decision maker's clients value having their contracts evenly spread, this might be an acceptable increase in costs to ensure that as many of the contracts' requirements as possible are fulfilled.

Summary
In this section the model was used to generate trade-off curves mapping the relationship between the two objectives of reducing operational costs and fairly evenly spreading contract pickups, allowing decision makers to choose an appropriate balance depending on profit margins and their clients needs.

8.2 Varying the demand to capacity ratio

In this section, we will examine how the cost per unit of a good is affected by increasing total demand for a planning horizon. We will also explore how such an increase affects the feasibility of the instance. From the example solution in Section 7.3 we saw that even though the total demand was at a 95% of total available capacity, which is a realistic demand to capacity ratio according to the case company, one of four voyages had underutilized stowage space.

The explanation for this is that not all contracts are loaded and stowed at the exact time; unloading goods in a port frees up space for loading new goods in the same port and in every port after that until all capacity is used again. In other words, the vessel capacity is a measure of the quantity that can be stowed on the vessel at a specific time during a voyage, not a measure of the total quantity of goods that can be transported in total over the entire voyage.

Below, we increase the total demand for a planning horizon in two different ways. First, in Section 8.2, we increase individual contract demands while keeping the number of contracts constant. Second, in Section 8.2, we increase the number of contracts, while keeping the contract demand for the already existing contracts constant. Both ways will increase total demand, but the different methods can affect feasibility in different ways by adding extra complexity to the model.

Increasing individual contract demand

In this section, we will increase individual contract demand equally, while keeping the number of contracts constant. This will increase the total demand transported and can decrease the cost per unit of a good. We created the instances required for this analysis by choosing four base instances, and then iteratively increasing the contract demand for each contract by 2%, saving the result of each iteration as a set of new instances. The specific instances used are AsE_10_100_3_3_DCR where $DCR \in \{94\%, 96\%, 97\%, 99\%, 101\%, 102\%, 104\%, 105\%, 107\%, 109\%, 111\%, 112\%, 114\%, 116\%, 118\%, 120\%, 122\%, 124\%, 126\%\}$.

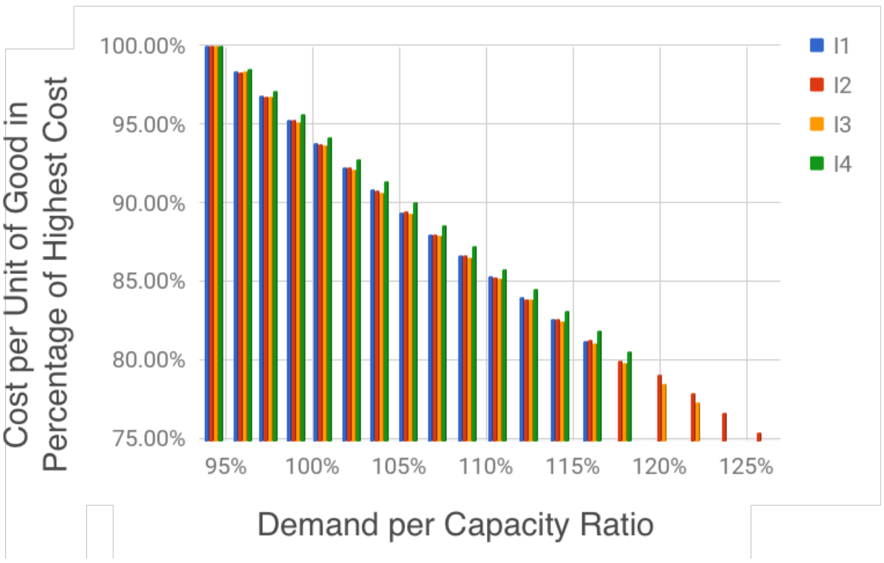


Figure 8.2: The effect of increasing demand on cost per unit of good (CPU) as a function of demand to capacity ratio (DCR). Vertical axis shows CPU as a percentage value of the CPU at 95% DCR, while the horizontal axes shows DCR.

In Figure 8.2, we have visually presented the effect of increasing demand on cost per unit of a good (CPU) as a function of demand to capacity ratio (DCR). The figure is a bar chart representing CPU as a percentage value of the CPU at 95% demand to capacity ratio. The bar chart contains one bar for each base instance, for each value of DCR in cases where a solution could be found. If a solution for a DCR level could not be found, the bar for the corresponding instances is not included. As an example, consider DCR value 126%, where the model was only able to find a feasible solution for instance two.

From Figure 8.2 we can see that cost per unit of good decreases at a similar rate for all four instances. There are however some slight differences. For example, instance four sees less improvement than the other instances. The explanation for these differences between instances is that in some cases the model has to reschedule contract pickups. As we increase demand for the instance, the solution to the instance before the increase may no longer be feasible. The model has to reschedule the contract pickups, which in those cases leads to an increase in operational costs. Observe also that demand can be increase to well above the 95% DCR, and first lead to infeasibility for instance one at 118% DCR, at 120% for instance four, and at 124% for instance three, while instance two became infeasible after 126% DCR.DCR.

Increasing the number of contracts

We created the instances required for this analysis by using four base instances and then iteratively increasing the number of contracts by two, by adding new contracts, saving the result of each iteration as a set of new instances. The specific instances used are AsE_10_C_3_3_DCR where $C \in [100, 180]$, with a step of two and DCR is dependent on C.

In Figure 8.3 we have visually presented the effect of increasing demand on cost per unit of a good (CPU) as a function of the number of contracts. The figure is a bar chart representing (CPU) as a percentage value of the (CPU) at 100 contracts. The bar chart contains one bar for each for base instance, for each number of contracts in cases where a solution could be found. If a solution for a contract level could not be found, the bar for the corresponding instances is not included. As an example, consider number of contracts value 150, where the model was only able to find a feasible solution for instance two and instance three.

From Figure 8.3 we can see results similar to those in Section 8.2. The CPU decreases at a similar rate for all four instances, with slight differences caused by solutions for a number of contracts level becoming infeasible as the number of contracts are increase and rescheduling is required. There is a larger difference between instances here in CPU. The explanation is that as new, randomly generated contracts are added, new transit time and evenly spread requirements are added to the instances. Contracts that have requirements that are not compatible with the already existing contracts force the model to reschedule to accommodate these requirements as best as possible, which is likely to increase cost. From Figure 8.4 we can see the demand to capacity

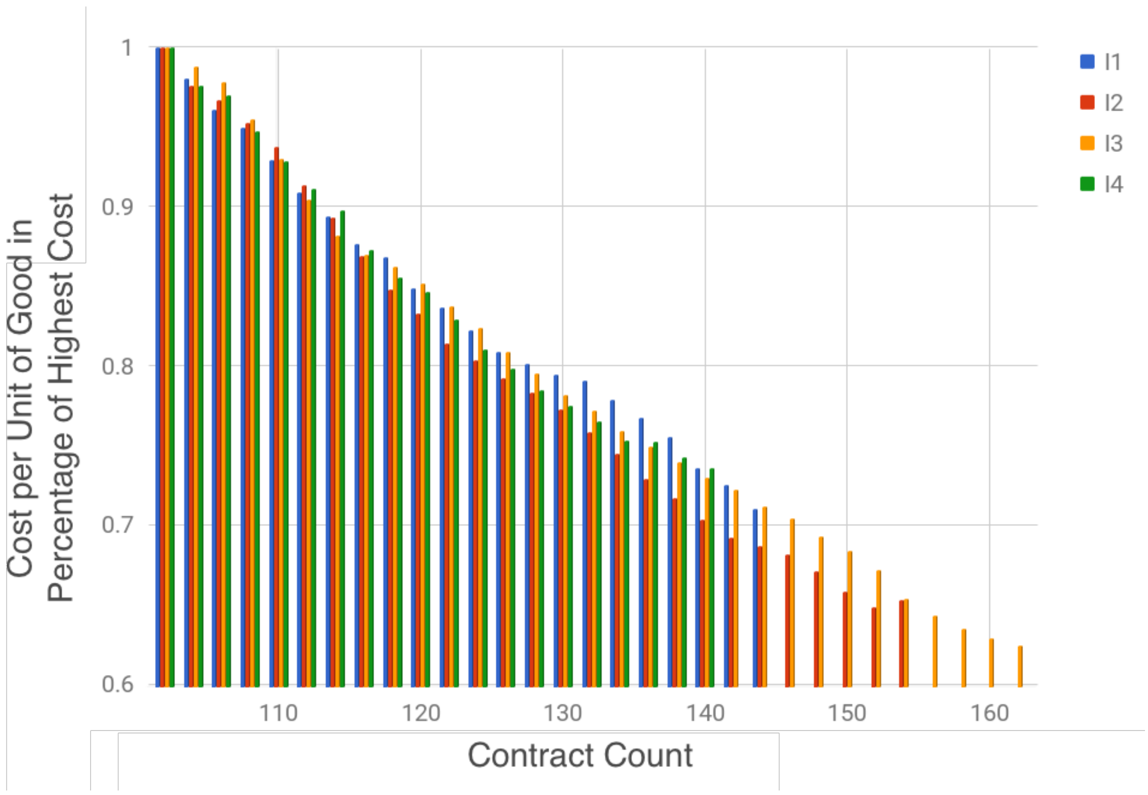


Figure 8.3: The effect of increasing demand on cost per unit of good (CPU) as a function of the number of contracts.. Vertical axis shows CPU as a percentage value of the CPU at at 100 contracts, while the horizontal axes shows the number of contracts.

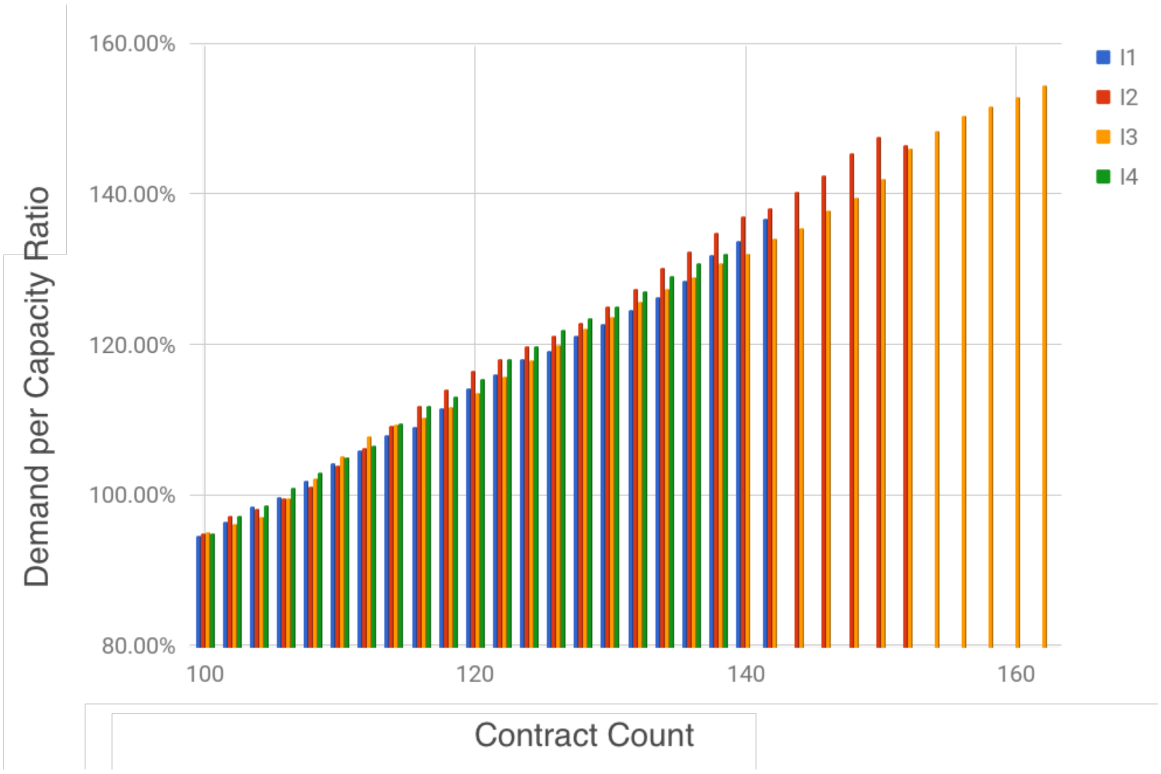


Figure 8.4: The effect of increasing demand on the demand to capacity ratio (DCR) as a function of the number of contracts. Vertical axis shows (DCR), while the horizontal axes shows the number of contracts.

ratio (DCR) as a function of the number of contracts. Observe that demand can be increase to well above the 95% DCR, and first lead to infeasibility for instance four at 132.0% DCR, at 137.7% for instance one, and at 146.5% for instance three, while instance two became infeasible after 154.4% DCR.

Analyzing the results

From Figure 8.2 and 8.4 we can see that increasing individual contract demand (method one) generally results in a smaller maximum increase in DCR than when adding more contracts (method two.) In fact, the average DCR where instances are no longer feasible for method one is 120%, while it is 142.4% for instances where we use method one. Hence we can increase it by 22.4% more on average when using method two. There are relatively large differences between instances in maximum DCR for each method, but based on the results above it seems like DCR can be increased more with method two than method one.

The explanation for this is that when increasing demand with method one we limit the possibility of rescheduling faster than with method two. For example, consider a set of contracts for a port p. The individual demand can only be increased until the total quantity to be picked up in that port exceeds the maximum available stowage capacity at p during the voyage with the least maximum available stowage capacity. There is some room for rescheduling, but if for example many of the contracts have a pickup frequency equal to the maximum number of voyages this port can become a bottleneck. This is also true for method two; however, when adding new demand we keep the demand per contract stable, which increases the possibilities of efficiently rescheduling to avoid bottlenecks. In addition, the new contracts can have requirements that are compatible with the requirements of the set of already existing contracts.

The results in this section indicates that by scheduling efficiently, decision makers can accept contracts for a total quantity of goods which takes demand to capacity ratio well above for example a common value of 95%. However, a model is only an approximation of reality. By blindly trusting the results of the model run, a vessel on a voyage may end up having to leave behind large quantities of goods on port visits as a result of overestimating available stowage space. Even if the cost of breaching contract is out-weighted by the reduced cost per unit of a good, this would not be a sustainable strategy, as clients are likely to value predictability. On the other hand, by being too conservative in our estimation to avoid risk, we can end up with a planning horizon with a lot of underutilized capacity.

In Chapter 4 we detailed how the model assumes that given the total base area x_p of all goods of each product type p, total real space use S can be predicted with a tolerable error from Equation 8.1,

$$S = F_1 x_1 + ... + F_p x_p, \tag{8.1}$$

and that the space utilization factors $F_1, ..., F_p$ can be estimated. A way of avoiding the risk that accompanies increasing the demand to capacity ratio would be to estimate the utilization factors and the corresponding error margins. This could be done, for example, by statistical analysis of historical data. To estimate these values is beyond this thesis. However, considering that a 10% increase in demand to capacity ratio can lower the cost per unit of good with approximately as much as 10%, such a study might be of interest to decision makers.

Summary
In this section it was demonstrated that through the models ability to estimate available stowage space, efficient routing and pickup scheduling could allow decision makers to accept more contracts, and increase the total demand to available capacity ratio above what is common practice, reducing costs per unit of good. Increasing the total demand does come at a risk of overestimating available stowage space and having to leave behind large quantities. It was outlined how estimating the utilization factors and the corresponding error margins could lessen some of the risk.

8.3 Removing voyages with low capacity utilization

In this section, we will examine if operational costs can be reduced by removing voyages with underutilized stowage capacity while keeping demand constant. From the example solution in Section 7.3 we saw that one of four voyages had underutilized stowage space. There are two reasons for a voyage with low capacity utilization to be scheduled at all by the model: either the vessels available does not have enough capacity to pick up all demand in less than the assigned number of voyages, or one or more contracts requires a pickup for every possible voyage during the planning horizon. Therefore, by decreasing the contract pickup frequency requirements of contracts that require the same number of pickups as the maximum number of voyages, we can run the model again to see if we can eliminate a voyage. If the solution still creates under utilized voyages, the process can be repeated until the instance is no longer feasible. However, by adjusting contract frequency requirements we break the terms of the adjusted contracts, which in turn means that we might be required to renegotiate these contracts. Any reduction in costs gained by removing one or more voyages that under utilize stowage capacity can be viewed as an approximation of the upper bound on the cost of renegotiating these contracts.

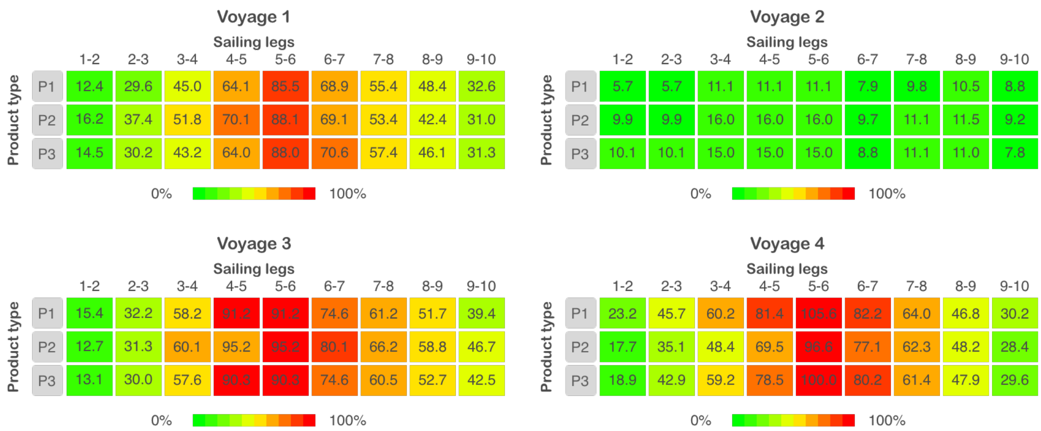


Figure 8.5: Heat map representation for instance from category group AsE_10_150_-4_4_95 where voyage two has low capacity utilization.

In this section, we will explore removal of a voyage from a planning horizon which has exactly one voyage with low capacity utilization. The instance is from the instance group AsE_10_150_4_4_95, with a penalty cost of 0.8; hence we are reducing a planning horizon with four voyages to a new planning horizon with three voyages. Figure 8.5 shows the capacity heat map representation for the original planning horizon. Observe that voyage two has a maximum peak capacity utilization of 16%, which makes it a good candidate for removal. In addition, we can see that voyage two is likely to have fewer port visits than the other voyages, as the capacity ratio remains constant over parts of the planning horizon. This can be seen directly in Figure 8.6.

By adjusting all contracts that have a maximum pickup frequency of four, we can run the model again to see if we can eliminate the voyage with underutilized stowage capacity. A heat map representation of the new planning horizon can be seen in 8.8, and we can observe from the figure that the model was able to remove the target voyage. Figure 8.6 and 8.7 shows network diagram representations of the old and new planning horizons respectively, with vessel speeds between ports. We can see that the voyages $V1$, $V2$ and $V3$ in the new planning horizon are topologically identical to voyages $V1$, $V3$ and $V4$ in the old planning horizon. A voyage by voyage comparison of operational costs between the two planning horizons is summarized in Table 8.1.

We can see that the total reduction in operational costs from removing one of four voyages is 20.8%. Of this, 20.2% comes from removing one voyage, eliminating sailings costs and port costs for that voyage, while the remaining 0.6% comes from a reduction

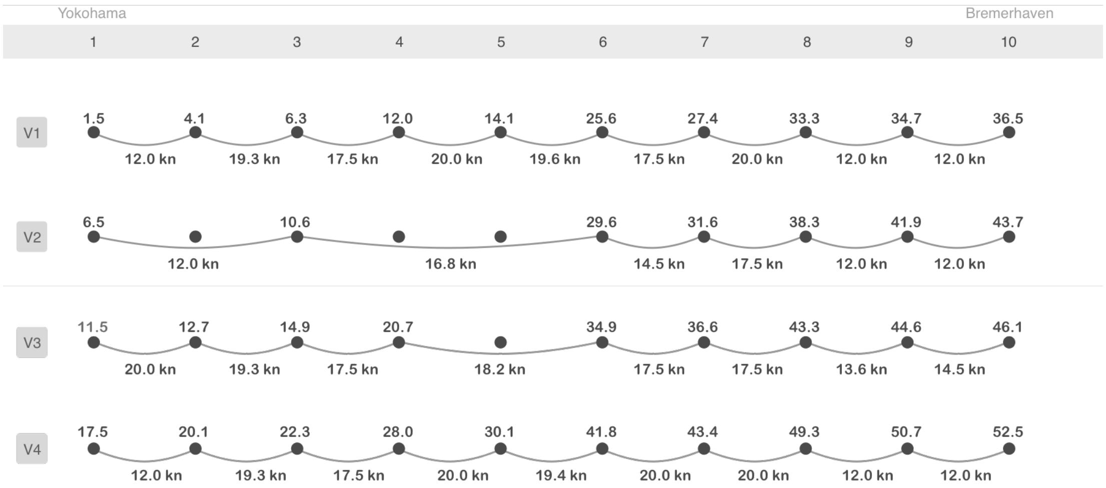


Figure 8.6: Diagram representation of an instance from instance group AsE_10_150_3_4_95, Showing arrival time in each port, as well as sailing speed between ports.

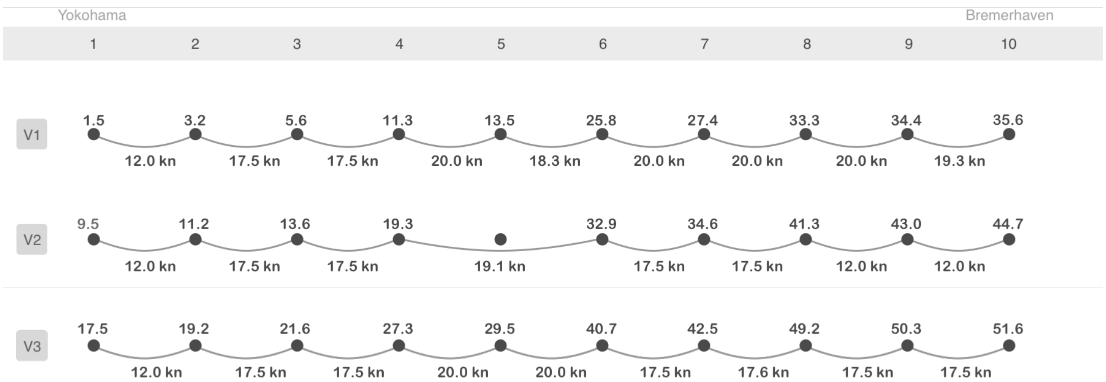


Figure 8.7: Diagram representation of an instance from instance group AsE_10_150_-3_4_95 where one voyage has been eliminated. Showing arrival time in each port, as well as sailing speed between ports.

in speed on other voyages. The voyage that was eliminated has fewer port stops than the other voyages. In addition, the vessel kept a generally lower sailing speed for that voyage, and therefore has lower sailing costs. As a consequence, $V2$ has a total cost that is lower than the other voyages. For example, the total cost is 71.3% lower than that of $V1$. Therefore, the total reduction in costs for the entire planning horizon is

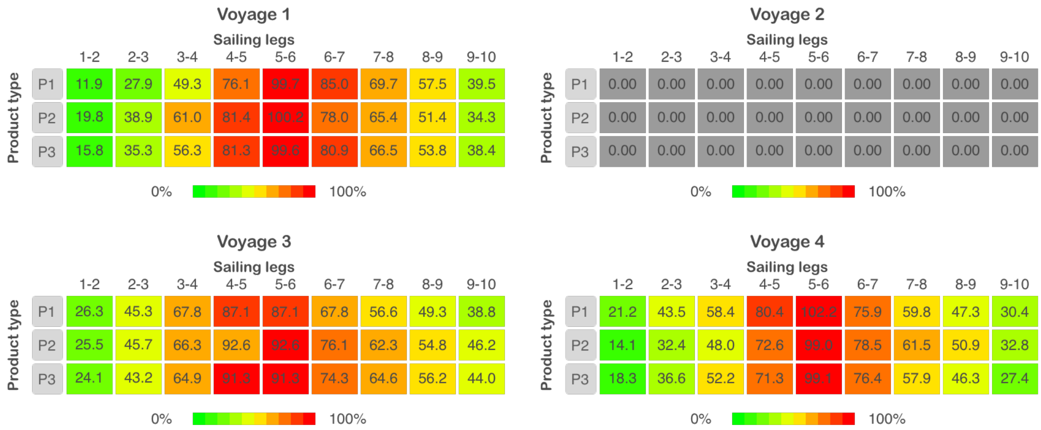


Figure 8.8: Heat map representation for instance from category group AsE_10_150_-4_4_95 where voyage two, a low capacity utilization, has been eliminated.

	Four Voyages		Δ Operational Costs		Three Voyages	
	Sailing	Port	Δ Sailing	Δ Port	New Sailing	New Port
Voyage 1	18.15%	10.20%	-0.30%	0.00%	17.85%	10.20%
Voyage 2	13.08%	7.14%	-13.08%	-7.14%	0.00%	0.00%
Voyage 3	13.92%	9.15%	0.07%	0.00%	13.99%	9.15%
Voyage 4	18.15%	10.20%	-0.30%	0.00%	17.85%	10.20%
Total	**63.31%**	**36.69%**	**-13.62%**	**-7.14%**	**49.68%**	**29.55%**
Operational costs: 100.00%				New operational cost: 79.24%		

Table 8.1: A summarized voyage by voyage comparison of operational costs between the original and reduced planning horizons.

not approximately 25% as one might expect when removing one of four voyages.

We can observe that while all port visits are removed from the eliminated voyage, $V2$, there are no changes in the number of port visits for voyages $V1$, $V3$ and $V4$. The reason for this can be seen in Figures 8.6 and 8.7: the ports visited on the eliminated voyage, $V2$, are all visited on every other voyage. This is as expected, as the contracts that were serviced on the eliminated voyage are contracts that require four pickups, and are therefore also serviced on every other voyage. Hence, no new port visits are necessary for completing the demand requirements of the contracts, the demand is simply shifted over to the other voyages.

In addition, we can see from the table that sailing costs have been reduced for $V1$ and $V4$, in addition to being completely eliminated for $V2$. The main reason for the reduction in $V1$ and $V4$ is that, with the reduction, the spreading requirements are easier to fulfill. There are now only two types of contracts with evenly spread requirements: those that can be scheduled for a maximum two pickups, and those that can be scheduled for a maximum three pickups. Contracts that get scheduled with three pickups have a spread time requirement of 9.3 ± 2, while those that have two pickups have a spread time requirement of 14 ± 2. For most contracts with two pickups for a given port, the model can simply schedule the pickups for $V1$ and $V3$, without breaking with those contracts that are scheduled for three pickups for the same port. This makes it easier for the model to schedule the contract pickups without deviation from evenly spread requirements, then when you also included contracts with four pickups. In Table 8.2 we can see the difference in contract deviation for each planning horizon. For the reduced planning horizon, fewer contracts deviate from evenly spread requirements, and deviation from evenly spread requirements per contract goes from 1.2 to 0.64.

Contract number	**Deviation time (days)** PH_4	**Deviation time (days)** PH_3
1	0.67	0.0
2	7.7	4.0
13	7.7	4.0
20	7.7	4.0
24	0.67	0.0
32	6.7	4.0
40	1.3	0.0
48	6.7	4.0
58	0.67	0.0
67	7.7	4.0
77	6.7	4.0
80	6.7	4.0
90	6.7	4.0
94	7.7	4.0
98	7.7	4.0
99	7.7	4.0
116	0.67	0.0
121	7.0	4.0
122	7.7	4.0
129	7.7	4.0
140	6.7	4.0

Table 8.2: All contracts that deviates from evenly spread restrictions, with deviation time in days, for the original planning horizon (PH_4) and the reduced planning horizon (PH_3).

Summary
In this section a voyage with low capacity utilization were removed from a planning horizon, in an effort to reduce costs. For our test instance, the total reduction in operational costs from removing one of four voyages is 20.8%. Of this, 20.2% comes from removing one voyage, eliminating sailings costs and port costs for that voyage, while the remaining 0.6% comes from a reduction in speed on other voyages. The process of removing one voyage also reduced deviation from evenly spread requirements. One suggestion for decision makers based on this analysis, is to limit the maximum contract pickup frequency to be less than the maximum number of possible voyages if possible. In practice, this would be equivalent to the process used for eliminating a voyage in this analysis, and would allow the model to remove unnecessary voyages when possible. In addition, this can possibly lessen deviation from evenly spread per contract, since it may reduce the number of contracts that have incompatible evenly spread requirements.

8.4 Merging planning horizons with same trade route

This thesis is done in cooperation with a case company that recently merged with another shipping company. Both companies transport goods over some of the same trade routes, with some differences in which ports are visited. In this section, we will examine the effects on costs when combining two planning horizons over the same trade route into one planning horizon, with the expected result of minimizing costs. One reason that costs might decrease may be that the model finds a solution where fewer voyages will be sailed than the sum of sailed voyages when the instances are run separately, which might decrease total sailing costs as well as port visit costs. By sailing fewer voyages we can also free up vessels which can be used to generate revenue elsewhere.

We created two instances to simulate the case outlined above, I1 and I2, belonging to instance group AsE_10_100_3_5_95, with a unit penalty cost of 0.8. To simulate that the voyages have differences in which ports they visit, we generated both instances from the same trade route, but each instance was limited to only visit seven of the ten available ports. The set of ports not visited were different for both instances. For each instance, a set of contracts, $\mathcal{C}^{I\#}$, was generated. In addition, for each instances a set of available vessels, $\mathcal{K}^{I\#}$, was sampled from the vessel pool. A third instance, I3, was created by combining I1 and I2, with a set of contracts $\mathcal{C}^{I1} \cup \mathcal{C}^{I2}$ and a set of available vessels $\mathcal{K}^{I1} \cup \mathcal{K}^{I2}$.

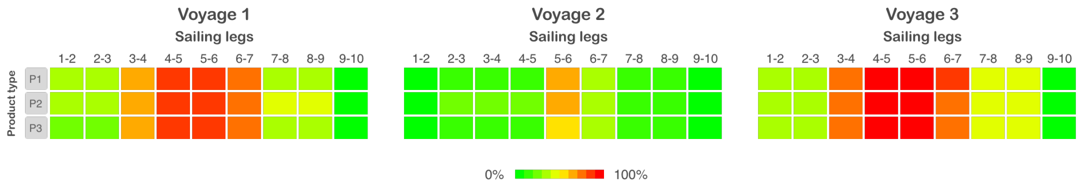


Figure 8.9: Heat map representation for instance I1 from category group AsE_10_-100_3_5_95, where voyage two has low capacity utilization.

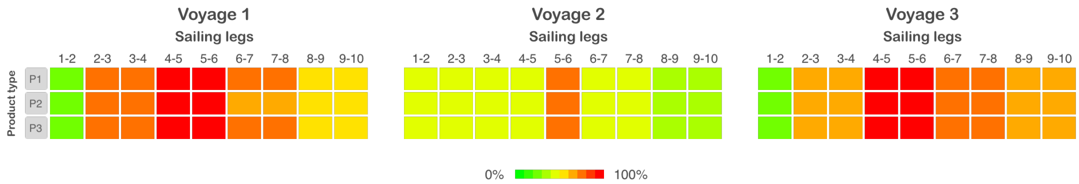


Figure 8.10: Heat map representation for instance I2 from category group AsE_10_-100_3_5_95, where voyage two has low capacity utilization.

The model was run with all three instances. We use a simplified heat map representation to visualize the results, were capacity percentages have been removed and capacity utilization is shown solely by colors. Observe from Figures 8.9 and 8.10 that I1 and I2, respectively, has one voyage with underutilized stowage capacity. We can see that voyage one and voyage three are near full capacity while voyage two is at approximately half capacity for both instances, all though slightly higher for I2.

Figure 8.11 shows the combination of I1 and I2's planning horizons, with a total maximum voyage number of six. We can see from the figure that two voyages have been eliminated, and that the remaining voyages have near full or full capacity utilization at peak capacity. The effects on the cost of this merger can be seen in Table 8.3.

The effects on the cost of this merger can be seen in Table 8.3. We can see from these results that merging the two planning horizons into one horizon has reduced total costs by roughly 17.7%. This needs some further explaining, as the reduction might seem lower than expected, given that two voyages were reduced. Sailing cost per voyage for the two separate voyages were $^{70.5\%}/_{6\ \text{voyages}} = 11.8\%$ of total planning horizon cost per voyage. For the merged planning horizon, however, sailing costs per voyage are $^{55.6\%}/_{4\ \text{voyages}} = 13.9\%$ of total planning horizon cost per voyage. This

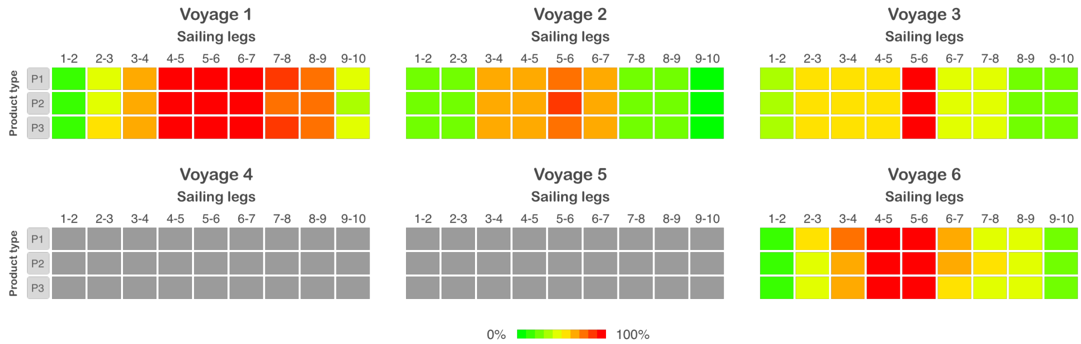


Figure 8.11: Heat map representation of the resulting merger of I1 and I2 planning horizons, with a total maximum voyage number of six. Two voyages has been eliminated.

	PH_1	PH_2	$\sum_{i\in\{1,2\}} PH_i$	PH_{merged}
Sailing cost	36.02%	34.44%	70.46%	55.55%
Port cost	14.79%	14.76%	29.55%	26.71%
Total	**50.81%**	**49.19%**	**100.0%**	**82.26%**

Table 8.3: The effects on cost of the merger of the two separate planning horizons (PH_1 and PH_2) to one, new planning horizon can be seen in Table

increase of sailings costs per voyage of 2.1% was caused partially by increases in vessel speeds to keep transit time requirements and to limit a deviation from evenly spread requirements, and partially by the model using larger more expensive vessels with larger to decrease the number of necessary voyages, and hence, port visits. Similarly, port cost per voyage for the two separate planning horizon was 4.9%, while port cost per voyage for the merged planning horizons 6.7%. The 1.8% increase in port cost per voyage is caused by the fact that the two separate planning horizons visited a different subset of the ten ports in the trade route. When the planning horizons were merged this increased the number of port visits which had to be performed per voyage. In addition, the two separate planning horizons had one voyage each with further port visits and lower sailing costs, which took the average down. In addition to reducing the total operational cost, merging the two planning horizons also freed up two vessels which can now be used to transport goods else where.

Summary

The thesis is done in cooperation with a case company that recently merged with another shipping company. Both companies transport goods over some of the same trade routes, with some differences in which ports are visited. In this section it was demonstrated that the model could be used to merge two planning horizons over the same trade route, and possibly reduce costs by eliminating voyages. For the specific instances used for this analysis, costs were reduced by 17.7%.

8.5 Relaxing strict transit time requirements

In this section, we will examine if operational costs can be reduced by making transit time requirements less strict. In Section 7.2 we saw that a large percentage of transit time constraints had abundant of transit time slack, and only a few transit time constraints were close to breaking their corresponding contracts' requirements. However, these few constraints, in combination with corresponding evenly spread constraints, indirectly enforced strict transit times on the other contract pickups. Together these constraints limit the model's ability to schedule contract pickups to exploit the transit time slack available. This leads to an interesting question, how would it affect operational costs if additional slack could be added to these contracts' transit time requirement.

The base instances used for this analysis are generated from the Asia to Europe trade route with ten ports. Each of the instances was selected from the instance group AsE_10_100_4_4_95. To create the instances with more lenient transit times, we increase the transit time required for contracts that had less than or equal to four days of transit time slack. We iteratively increased it with one, two, three, and four days for all instances, creating in total 20 additional instances. Results are aggregated and summarized in Table 8.4, which shows the average improvements of operational costs in percentage compared to the solution with no additional slack, for one, two, three and four additional days. The run time limit for this analysis is 1800 seconds.

Contracts	One day	Two days	Three days	Four days
50	0.36%	0.70%	1.01%	1.29%
100	0.37%	0.72%	1.06%	1.36%
150	0.39%	0.76%	1.09%	1.41%

Table 8.4: Reductions in cost increases as a function of the number of contracts and the number of additional slack days added to the strictest transit time requirement.

We can see from Table 8.4 that reductions in cost increase both with the number of contracts, and the number of additional slack days. All though the reductions are relatively small, the magnitude of the costs are in turn very large.

Our instance generator can generate every possible transit time between the minimum and the maximum time it can take to sail between two ports, plus an additional slack which in our cases was four days independent of distance. Further analysis of individual solutions of the instances run for this analysis shows that the transit times that benefited operational cost the most when relaxed, were those of larger distances. This suggests that our generator could probably be improved by making slack time distance dependent, as two ports that are far apart are more likely to have many ports between them meaning that a vessel might need more slack for that distance. The model helped to identify this potential problem with our instance generator. Similarly, the model could be used by decision makers to identify lower limits for transit times between ports, to reduce operational costs.

Summary
In this section it was demonstrated how the model could be used to learn new, applicable knowledge, by studying the effects on operational cost of relaxing strict transit time requirements. The model could be used by decision makers to identify lower limits for transit times between ports, to reduce operational costs.

8.6 Summary

In this chapter, we analyzed solutions for selected instances, with the goal of exploring practical applications and gaining managerial insight that can be helpful to decision makers. The model was used to generate trade-off curves mapping the relationship between the two objectives of reducing operational costs and fairly evenly spreading contract pickups, allowing decision makers to choose an appropriate balance depending on profit margins and their clients needs. Several ways of reducing costs was demonstrated. It was shown that through the models ability to estimate available stowage space, efficient routing and pickup scheduling could allow decision makers to accept more contracts, and increase the total demand to available capacity ratio above what is common practice, reducing costs per unit of good. Several use cases were shown where costs could be reduced by eliminating voyages with poor capacity utilization to reduce costs and free up vessels for use else were. It was also demonstrated how the model could be used to learn new, applicable knowledge, by studying the effects on operational cost of relaxing strict transit time requirements.

Chapter 9

Concluding Remarks

The Single Trade Routing Problem (STRP) for RoRo shipping set out to plan routes for several voyages over a planning horizon, as well as plan for which contracts to service and which quantities to load and unload at scheduled port visits. These are important operational decisions made by RoRo operators with the goal of gaining competitive advantage in a marketplace with low demand and excess capacity. A good decision support system is essential to help managers make informed decisions, reduce cost and eliminating waste. The purpose of this thesis have been to develop a better understanding of the STRP and gain valuable insights for decision makers, and is motivated by the conviction that a good decision support system has the potential for significant benefits also to RoRo providers in maritime transportation.

The aim of the work has been to create a model which reflects the real problem with a practical degree of accuracy. A mathematical formulation of The Single Trade Routing Problem has been further developed, by introducing new and comprehensive extensions. The model take into account variable vessel speeds, a heterogeneous fleet where vessels have different capacities and fuel usage, available stowage space estimation, and differences in capacity and space utilization for different decks within a single vessel. The formulation also include important features such as contract pickup frequency requirements, fairly evenly spread requirements for contract pickups, transit time requirements, and minimum and maximum loading quantity requirements per pickup.

A commercial MIP solver was used to explore the difficulty of solving instances with the model. The introduction of transit time requirements to contracts drastically increased model complexity, and caused performance in terms of run time to worsen. While the introduction of evenly spread constraints to some degree countered this increase, the results of the computational study show that run time increases non linearly for all important dimensions, and that solving instances larger than the relatively small test instances used in this thesis within a reasonable time limit is not possible.

A detailed study of instance solutions were performed, to gain insight about the problem and study how sensible the solutions are in relation to the real problem. A further study into practical application of the model was undertaken to gain managerial insight that can be helpful to decision makers. The model was used to generate trade-off curves mapping the relationship between the two objectives of reducing operational costs and fairly evenly spreading contract pickups, allowing decision makers to choose an appropriate balance depending on profit margins and their clients needs. Several ways of reducing costs was demonstrated. It was shown that through the models ability to estimate available stowage space, efficient routing and pickup scheduling could allow decision makers to accept more contracts, and increase the total demand to available capacity ratio above what is common practice, reducing costs per unit of good. Several use cases were shown where costs could be reduced by eliminating voyages with poor capacity utilization to reduce costs and free up vessels for use else were. It was also demonstrated how the model could be used to learn new, applicable knowledge, by studying the effects on operational cost of relaxing strict transit time requirements.

Chapter 10

Future Research

The results from the computational study show that run time increase non linearly for all dimensions and that solving instances larger than relatively small test instances used in this thesis within a reasonable time limit is not possible. For further research, a natural next step would, therefore, be to develop a heuristic solver for The Single Trade Routing Problem.

The model could also be extended to better reflect the real problem. Port costs are now constant but could be extended to reflect a more realistic situation where they are dependent on factors that affect time in port, like for example which product types and which quantities are loaded. In our model vessels travels unidirectionally through a trade route. This limits the quantity vessels can transport, by limiting the ability to visit adjacent ports in an optimal order. In addition, it put limits on how to solve evenly spread and transit time restrictions. Even though this would add complexity to the model, the number of additional arcs added to the network from allowing bi-directional sailing could be limited with the use of domain knowledge and knowledge about specific trade routes. A robustness study of the model is also interesting. Real-world problems usually have large variations in input variables, that may in our case depend on factors like weather, which can effect sailing speeds and arrival time, and random events at a port, which may cause delays. The model as it is now assumes constancy in all aspects.

In addition, we find the extension to solving multiple trade routes interesting, as scheduling for a single trade route would be dependent on scheduling for other trade routes. It is possible to single trade route instances separately. Using solution values for one or more instance as input in an instance that follows it geographically and in time can be done by planners manually. However, this would most likely not be an optimal solution as the optimal solutions to some local single trade route instances might not be the best solution for global optimality.

Further more, we find a study using real data interesting. Especially in relation to estimating utilization factors to limit the risk of over and under stowing, and as a way to facilitate better capacity utilization and decreasing the cost per unit of a good. Our model assumes a linear relationship between products spatial attributes and the real space it takes when stowed, that when given the total base area x_p of all goods of each product type p, total real space used can be predicted with a tolerable error. If this assumption holds utilization factors could be estimated using multiple linear regression over historical data. If the relationship is more complex, a heuristic solver could allow for more complex and accurate nonlinear statistical methods to be used.

Bibliography

Agarwal, R. & Ergun, Ö. (2008). Ship scheduling and network design for cargo routing in liner shipping. *Transportation Science*, *42*(2), 175–196.

Alvarez, J. F. (2009). Joint routing and deployment of a fleet of container vessels. *Maritime Economics & Logistics*, *11*(2), 186–208.

Álvarez, J. F. (2012). Mathematical expressions for the transit time of merchandise through a liner shipping network. *Journal of the Operational Research Society*, *63*(6), 709–714.

Ambrosino, D., Sciomachen, A., & Tanfani, E. (2004). Stowing a containership: the master bay plan problem. *Transportation Research Part A: Policy and Practice*, *38*(2), 81–99.

Andersson, H., Christiansen, M., & Fagerholt, K. (2011a). The maritime pickup and delivery problem with time windows and split loads. *INFOR: Information Systems and Operational Research*, *49*(2), 79–91.

Andersson, H., Duesund, J. M., & Fagerholt, K. (2011b). Ship routing and scheduling with cargo coupling and synchronization constraints. *Computers & Industrial Engineering*, *61*(4), 1107–1116.

Andersson, H., Fagerholt, K., & Hobbesland, K. (2015). Integrated maritime fleet deployment and speed optimization: case study from roro shipping. *Computers & Operations Research*, *55*, 233–240.

Avriel, M., Penn, M., Shpirer, N., & Witteboon, S. (1998). Stowage planning for container ships to reduce the number of shifts. *Annals of Operations Research*, *76*, 55–71.

Belanger, N., Desaulniers, G., Soumis, F., & Desrosiers, J. (2006). Periodic airline fleet assignment with time windows, spacing constraints, and time dependent revenues. *European Journal of Operational Research*, *175*(3), 1754–1766.

Brønmo, G. [G], Christiansen, M., & Nygreen, B. (2007). Ship routing and scheduling with flexible cargo sizes. *Journal of the Operational Research Society*, *58*(9), 1167–1177.

Brønmo, G. [Geir], Nygreen, B., & Lysgaard, J. (2010). Column generation approaches to ship scheduling with flexible cargo sizes. *European Journal of Operational Research*, *200*(1), 139–150.

Bunker, S. (2017). Rotterdam bunker prices. Accessed: 2017-06-13. Retrieved from https://shipandbunker.com/prices/emea/nwe/nl-rtm-rotterdam#IFO380

Chen, C. & Zeng, Q. (2010). Designing container shipping network under changing demand and freight rates. *Transport*, *25*(1), 46–57.

Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D. (2007). Maritime transportation. *Handbooks in operations research and management science*, *14*, 189–284.

Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D. (2013). Ship routing and scheduling in the new millennium. *European Journal of Operational Research*, *228*(3), 467–483.

Christiansen, M., Fagerholt, K., & Ronen, D. (2004). Ship routing and scheduling: status and perspectives. *Transportation science*, *38*(1), 1–18.

Christiansen, M. & Nygreen, B. (1998). A method for solving ship routing problemswith inventory constraints. *Annals of Operations Research*, *81*, 357–378.

Chu, C.-W., Kuo, T.-C., & Shieh, J.-C. (2003). A mixes integer programming model for routing containerships. *Journal of Marine Science and Technology*, *11*(2), 96–103.

Chuang, T.-N., Lin, C.-T., Kung, J.-Y., & Lin, M.-D. (2010). Planning the route of container ships: a fuzzy genetic approach. *Expert Systems with Applications*, *37*(4), 2948–2956.

Fagerholt, K., Hvattum, L. M., Johnsen, T. A., & Korsvik, J. E. (2013). Routing and scheduling in project shipping. *Annals of Operations Research*, *207*(1), 67–81.

Fagerholt, K., Johnsen, T. A., & Lindstad, H. (2009). Fleet deployment in liner shipping: a case study. *Maritime Policy & Management*, *36*(5), 397–409.

Fagerholt, K., Laporte, G., & Norstad, I. (2010). Reducing fuel emissions by optimizing speed on shipping routes. *Journal of the Operational Research Society*, *61*(3), 523–529.

Fischer, A., Nokhart, H., Olsen, H., Fagerholt, K., Rakke, J. G., & Stålhane, M. (2016). Robust planning and disruption management in roll-on roll-off liner shipping. *Transportation Research Part E: Logistics and Transportation Review*, *91*, 51–67.

Gelareh, S., Nickel, S., & Pisinger, D. (2010). Liner shipping hub network design in a competitive environment. *Transportation Research Part E: Logistics and Transportation Review*, *46*(6), 991–1004.

Halvorsen-Weare, E. E., Fagerholt, K., Nonås, L. M., & Asbjørnslett, B. E. (2012). Optimal fleet composition and periodic routing of offshore supply vessels. *European Journal of Operational Research*, *223*(2), 508–517.

Hennig, F., Nygreen, B., Furman, K. C., Song, J., & Kocis, G. R. (2011). Crude oil tanker routing and scheduling. *INFOR: Information Systems and Operational Research*, *49*(2), 153–170.

Hwang, H.-S., Visoldilokpun, S., & Rosenberger, J. M. (2008). A branch-and-price-and-cut method for ship scheduling with limited risk. *Transportation science*, *42*(3), 336–351.

IMO. (2016). International maritime organization. Accessed: 2016-12-17. Retrieved from https://business.un.org/en/entities/13

International Chamber of Shipping, I. (2016). Shipping facts. Retrieved from http://www.ics-shipping.org/shipping-facts/shipping-facts

Jetlund, A. S. & Karimi, I. (2004). Improving the logistics of multi-compartment chemical tankers. *Computers & Chemical Engineering*, *28*(8), 1267–1283.

Johnson, D. S., Lenstra, J. K., & Kan, A. (1978). The complexity of the network design problem. *Networks*, *8*(4), 279–285.

Jung, J. U., Kang, M. H., Choi, H. R., Kim, H. S., Park, B. J., & Park, C. H. (2011). Development of a genetic algorithm for the maritime transportation planning of car carriers. In *Dynamics in logistics* (pp. 481–488). Springer.

Kang, M. H., Choi, H. R., Kim, H. S., & Park, B. J. (2012). Development of a maritime transportation planning support system for car carriers based on genetic algorithm. *Applied Intelligence*, *36*(3), 585–604.

Kobayashi, K. & Kubo, M. (2010). Optimization of oil tanker schedules by decomposition, column generation, and time-space network techniques. *Japan Journal of Industrial and Applied Mathematics*, *27*(1), 161–173.

Korsvik, J. E. & Fagerholt, K. (2010). A tabu search heuristic for ship routing and scheduling with flexible cargo quantities. *Journal of Heuristics*, *16*(2), 117–137.

Korsvik, J. E., Fagerholt, K., & Laporte, G. (2011). A large neighbourhood search heuristic for ship routing and scheduling with split loads. *Computers & Operations Research*, *38*(2), 474–483.

Lawrence, S. A. (1972). *International sea transport: the years ahead*. Lexington Books.

Li, C.-L. & Pang, K.-W. (2011). An integrated model for ship routing and berth allocation. *International Journal of Shipping and Transport Logistics*, *3*(3), 245–260.

Lin, D.-Y. & Liu, H.-Y. (2011). Combined ship allocation, routing and freight assignment in tramp shipping. *Transportation Research Part E: Logistics and Transportation Review*, *47*(4), 414–431.

Lindstad, H., Asbjørnslett, B. E., & Strømman, A. H. (2011). Reductions in greenhouse gas emissions and cost by shipping at lower speeds. *Energy Policy*, *39*(6), 3456–3464.

Logistics, W. W. (2016a). Roro – the safer, smarter way. Accessed: 2016-10-17. Retrieved from http://www.2wglobal.com/global-network/fleet/fleet-overview/interactive-vessel/

Logistics, W. W. (2016b). Take a look inside. Accessed: 2016-10-17. Retrieved from http://www.2wglobal.com/news-and-insights/infographics/take-a-look-inside/

Lu, H.-A. (2002). Modelling ship. *Journal of Marine Science and Technology*, *10*(1), 61–67.

Meng, Q., Wang, S., Andersson, H., & Thun, K. (2013). Containership routing and scheduling in liner shipping: overview and future research directions. *Transportation Science*, *48*(2), 265–280.

Meng, Q. & Wang, T. (2011b). A scenario-based dynamic programming model for multi-period liner ship fleet planning. *Transportation Research Part E: Logistics and Transportation Review*, *47*(4), 401–413.

Norstad, I., Fagerholt, K., Hvattum, L. M., Arnulf, H. S., & Bjørkli, A. (2015). Maritime fleet deployment with voyage separation requirements. *Flexible Services and Manufacturing Journal*, *27*(2-3), 180–199.

Norstad, I., Fagerholt, K., & Laporte, G. (2011). Tramp ship routing and scheduling with speed optimization. *Transportation Research Part C: Emerging Technologies*, *19*(5), 853–865.

Øvstebø, B. O., Hvattum, L. M., & Fagerholt, K. (2011a). Optimization of stowage plans for roro ships. *Computers & Operations Research*, *38*(10), 1425–1434.

Øvstebø, B. O., Hvattum, L. M., & Fagerholt, K. (2011b). Routing and scheduling of roro ships with stowage constraints. *Transportation Research Part C: Emerging Technologies*, *19*(6), 1225–1242.

Pang, K.-W., Xu, Z., & Li, C.-L. (2011). Ship routing problem with berthing time clash avoidance constraints. *International Journal of Production Economics*, *131*(2), 752–762.

Papadakis, N. A. & Perakis, A. N. (1989). A nonlinear approach to the multiorigin, multidestination fleet deployment problem. *Naval Research Logistics (NRL)*, *36*(4), 515–528.

Patricksson, Ø. S., Fagerholt, K., & Rakke, J. G. (2015). The fleet renewal problem with regional emission limitations: case study from roll-on/roll-off shipping. *Transportation Research Part C: Emerging Technologies*, *56*, 346–358.

Perakis, A. & Jaramillo, D. (1991). Fleet deployment optimization for liner shipping part 1. background, problem formulation and solution approaches. *Maritime Policy and Management*, *18*(3), 183–200.

Reinhardt, L. B. & Pisinger, D. (2012). A branch and cut algorithm for the container shipping network design problem. *Flexible Services and Manufacturing Journal*, *24*(3), 349–374.

Ronen, D. (2011). The effect of oil price on containership speed and fleet size. *Journal of the Operational Research Society*, *62*(1), 211–216.

Sea-Distance. (2017). Sea distance. Accessed: 2017-02-10. Retrieved from https://sea-distances.org/

Shintani, K., Imai, A., Nishimura, E., & Papadimitriou, S. (2007). The container shipping network design problem with empty container repositioning. *Transportation Research Part E: Logistics and Transportation Review*, *43*(1), 39–59.

Sigurd, M. M., Ulstein, N. L., Nygreen, B., & Ryan, D. M. (2005). Ship scheduling with recurring visits and visit separation requirements. In *Column generation* (pp. 225–245). Springer.

Takano, K. & Arai, M. (2011). Study on a liner shipping network design considering empty container reposition. *Journal of the Japan Society of Naval Architects and Ocean Engineers*, *13*, 175–182.

Ting, S.-C. & Tzeng, G.-H. (2003). Ship scheduling and cost analysis for route planning in liner shipping. *Maritime Economics & Logistics*, *5*(4), 378–392.

UNCTAD. (2015). *Review of maritime transport*. United Nations Conference on Trade and Development. United Nations publication.

UNCTAD. (2016). *Review of maritime transport*. United Nations Conference on Trade and Development. United Nations publication.

Vallestad, T. & Weggersen, A. (2016, December). *The single trade routing problem in roro shipping* (Master's thesis, Norwegian University of Science and Technology).

Wang, S. & Meng, Q. (2011). Schedule design and container routing in liner shipping. *Transportation Research Record: Journal of the Transportation Research Board*, (2222), 25–33.

Yao, Z., Ng, S. H., & Lee, L. H. (2012). A study on bunker fuel management for the shipping liner services. *Computers & Operations Research*, *39*(5), 1160–1172.