Sofie Smith Vågen
Jørgen Bjaarstad Nikolaisen

A Combined Optimization-Simulation Framework for a Maritime Inventory Routing Problem Under Uncertainty

Master's thesis in Industrial Economics and Technology Management
Supervisor: Peter Schütz
June 2019

Master's thesis

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

NTNU
Kunnskap for en bedre verden

Hydro

Sofie Smith Vågen
Jørgen Bjaarstad Nikolaisen

A Combined Optimization-Simulation Framework for a Maritime Inventory Routing Problem Under Uncertainty

Master's thesis in Industrial Economics and Technology Management
Supervisor: Peter Schütz
June 2019

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

NTNU
Norwegian University of Science and Technology

Purpose of Master's Thesis

This Master's thesis purpose is to develop a solution framework and mathematical model(s) to a maritime inventory routing problem faced with uncertainty. The problem considers industrial shipping of raw material from South America and Europe to plants along the Norwegian coastline. Both the times for when ships begin loading in ports and the travel times between loading and unloading ports are considered uncertain in this thesis.

The goal is to find shipping schedules that perform well under this uncertainty. To do this, an iterative solution framework combining optimization and simulation will be developed and implemented. In the first stage an optimization model finds candidate solutions that will be evaluated in the second stage using a simulation model. The results of the simulations are then used to update the original problem and generate new solutions.

Preface

This Master's thesis is the concluding part of our Master of Science in Industrial Economics and Technology Management at the Norwegian University of Science and Technology (NTNU). Our field of specialization is Managerial Economics and Operations Research at the Department of Industrial Economics and Technology Management.

The Master's thesis was written during the spring semester of 2019 and is based on our work in the specialization project during the fall semester of 2018 (Nikolaisen and Vågen 2018). The goal of the thesis is to find shipping schedules for a maritime inventory routing problem faced by Norsk Hydro ASA with uncertainty in both travel times and the port admission times in loading ports.

We would like to thank our academic supervisor Peter Schütz for valuable discussions, excellent feedback and guidance throughout the project. Furthermore, we would like to thank Norsk Hydro ASA for their cooperation, especially Sindre Bolseth for his insights and valuable input.

Trondheim, June 11, 2019

Sofie Smith Vågen

Jørgen Bjaarstad Nikolaisen

Abstract

This Master's thesis presents an optimization-simulation framework for a tactical Maritime Inventory Routing Problem under uncertainty faced by a global aluminium producer. The aim is to find shipping schedules performing well under uncertain travel and port admission times. To accomplish this we develop an iterative feedback framework combining simulation with two optimization models.

The producer's decisions are divided into loading- and ship selection decisions, taken weeks prior to departure, and routing- and unloading decisions that are more flexible and can be altered after the uncertainty is revealed. Stage 1 of the solution framework generates planned schedules where the loading- and ship selection decisions are made. Stage 2 evaluates the solutions using simulation with a re-routing model to allow for some flexibility in the routing- and unloading decisions. The estimated cost of uncertainty for the planned schedule is then associated with the stage 1 solution in the next iteration of the framework, prompting the stage 1 model to create a different schedule. This process is repeated until it terminates with an estimated optimal solution under uncertainty.

The framework has a two-stage approach; revealing all the uncertain parameters between stage 1 and 2. However, to better imitate the real-world information structure, we further develop a multistage extension to the re-routing model. This model aims to capture the challenges of re-routing under incomplete information. Further, we implement several improvements to reduce the framework's runtime. Most notably, we extend the framework to a two-phase evaluation approach as well as extract multiple solutions in each iteration and evaluate them in parallel.

The results of our computational study show that the solutions found using the framework perform significantly better under uncertainty than the optimal deterministic solutions. Also, the framework with multistage information structure produces more robust solutions with respect to uncertainty in travel and port admission times than the original two-stage approach. Further, the multistage framework can be solved within a reasonable runtime for realistic problem instances.

vi

Sammendrag

Denne masteroppgaven presenterer et rammeverk som kombinerer optimering og simulering for å løse et maritimt ruteplanleggings- og lagerstyringsproblem (MIRP) under usikkerhet for en global aluminiumsprodusent. Hensikten er å konstruere en leveringsplan over en taktisk planleggingshorisont som presterer godt under usikkerhet i reisetid og ankomsttid i lastehavner. For å oppnå dette utvikler vi et iterativt rammeverk som kombinerer simulering med to optimeringsmodeller.

Produsentens beslutninger deles i beslutninger som må tas flere uker før avreise; valg av skip og lastehavn, og beslutninger som er mer fleksible; ruting av skip og lossing. Steg 1 av rammeverket genererer en planlagt leveringsplan der valg av skip og lastehavn besluttes endelig. Steg 2 bruker simulering med en rerutingsmodell til å evaluere de planlagte leveringsplanene der rerutingsmodellen tillater fleksibilitet i ruting- og lossebeslutningene. En estimert usikkerhetskostnad assosieres med den planlagte leveringsplanen, noe som fører til at steg 1 av modellen genererer en ny leveringsplan i neste iterasjon av rammeverket. Denne prosessen gjentas til rammeverket har funnet en planlagt leveringsplan som er optimal under usikkerhet.

Rammeverket har initielt en tostegs-tilnærming der alle de usikre parametrene avsløres samtidig mellom steg 1 og steg 2. Vi utvikler videre en rerutingsmodell med flerstegs-tilnærming for å nærmere etterligne den virkelige beslutningsprossen. Målet med denne utvidelsen er å fange opp utfordringene ved å planlegge rerutingen med mangelfull informasjon. Deretter implementerer vi flere tiltak for å redusere kjøretiden inkludert parallel prosessering av løsninger og en to-fasetilnærming for evalueringen av løsnigner.

Resultatene viser at leveringsplanene som genereres av vårt rammeverk presterer betydelig bedre under usikkerhet enn de deterministik optimale leveringsplanene. Videre produserer rammeverket med flerstegs-tilnærming løsninger som håndterer usikkerheten i lastetid og reisetid bedre enn løsningene fra det originale rammeverket. Rammeverket løser realistiske probleminstanser innen en rimelig tid med både tostegs- og flerstegs-tilnærming.

viii

Contents

xiv

List of Tables

List of Figures

Chapter 1

Introduction

Aluminium is suitable for a great variety of products due to its unique combination of favourable properties, such as formability, ductility, easiness to recycle, conductivity, corrosion resistance and its alloys' strength. Hence, the aluminium industry has experienced a significantly growth during the past decades with a cumulative annual growth rate of 5.5% in global demand from 2000 to 2017.

Primary aluminium is produced from alumina, a fine powder. The largest alumina exporters are South America and Oceania, while large net importers are Europe and North America. The great distance between exporters and importers along with the large amount of product transported, make maritime transportation and logistics an essential part of aluminium's value chain.

The Norwegian aluminium company Norsk Hydro ASA, henceforth referred to as Hydro, is involved in operations across the whole value chain. In 2017 it was the ninth largest aluminium producer globally, with operations across five continents. The aluminium plants are mostly located in Norway, while alumina is refined in Brazil. In addition, external alumina producers cover a minor portion of Hydro's alumina demand. Ships are used to transport alumina to the Norwegian aluminium plants. Due to fierce competition from producers located in low cost countries, the incentive to create a maritime transportation plan as cost efficient as possible is high.

Today, the team creating the tactical plans for maritime transportation in Hydro has few supportive tools for decision making in the planning process. Hydro needs to consider inventory management of the raw material at the aluminium plants. Hence, the problem can be classified as a maritime inventory routing problem, hereafter referred to as MIRP.

The scheduling decisions at the alumina ports must be taken 3-4 weeks prior to loading when Hydro makes a call-off on the shipping contracts. However, the decisions regarding routing and scheduling to and between the aluminium plants are flexible up until a few days before arrival at the aluminium plants. To further complicate the process, the travel time and time to begin loading in the alumina ports are uncertain. Therefore, estimating the time of arrival at the aluminium plants is difficult. This often causes rerouting of the ships and unplanned split deliveries between the unloading ports to avoid stock outs.

In this thesis, the goal is to develop an optimization-simulation framework to identify the best planned schedules when uncertainty is taken into consideration. To this end, an iterative feedback framework consisting of two stages is developed. It tries to utilize the difference between call-off and routing decisions by locking the call-off decisions in stage 1, and deciding the flexible routing decisions in stage 2. Both stages include a MIRP optimization model, with the purposes of generating planned schedules and re-routing after the uncertainty has been realized, respectively.

The framework iterates between stage 1 and stage 2. Stage 1 generates planned schedules, called potential solution. They contain both robust and flexible decisions, where the latter can change in the evaluation process. Stage 2's purpose is to evaluate the potential solutions by using a simulation model with the re-routing model to determine the flexible decisions after the scenarios are generated.

If a potential solution experiences stock-outs in a scenario, rendering the rerouting model infeasible, a high stock-out cost occurs. The difference between the realized costs in stage 2 and the planned costs in stage 1 is used to estimate a cost of uncertainty for a potential solution. Then, the cost of uncertainty is used as feedback to the stage 1 model, such that the potential solution is coupled with its cost of uncertainty in the remaining iterations. The framework terminates with an estimated optimal solution when the cost of uncertainty added to the planned costs of a stage 1 solution is better than the planned cost of any other solutions.

In order to better imitate the real-world information structure, we implement a multistage extension to the process of revealing information in the re-routing model. This model aims to capture the challenges of planning under incomplete information by revealing the uncertain parameters for different ships at different times. Furthermore, a two-phase evaluation approach is developed to get more reliable estimates of the performance of the potential solutions under uncertainty.

We also seek to evaluate the performance of the framework both with respect to runtime and solution quality. To further reduce the runtime, we present several initiatives. Moreover, the effect of including the extensions is also explored in detail. Finally, the performance and characteristics of the solutions suggested by the complete framework are analyzed.

The contribution of this Master's thesis to the literature is twofold. To the best of our knowledge, we are the first to develop an iterative feedback approach to solving the MIRP under uncertainty. Due to the complexity of the deterministic problem, it is often challenging to solve realistic instances when considering uncertainty. Our framework is able to solve realistic instances within a reasonable runtime while considering both robust and flexible decisions. Secondly, we introduce multistage information structure and decision making without increasing the runtime of the framework dramatically.

The Master thesis is arranged as follows: Firstly, the relevant background information is presented in chapter 2. Thereafter, chapter 3 presents a problem description. A review of related literature is given in chapter 4. Then, the chapters 5 and 6 present the thesis' optimization-simulation framework and some further extensions, respectively. Chapter 7 is a case study presenting the input data used for the analysis and the underlying assumptions. The following chapter 8 provides a computational study where the computational time and the performance of the framework and its extensions are discussed. Lastly, concluding remarks and some interesting topics for future research are presented in chapter 9.

Chapter 2

Background

In this chapter we present the relevant background information and context for the rest of this Master's thesis. Firstly, an overview of the global aluminium industry is provided in section 2.1. Secondly, the aluminium producer Hydro and its operations are presented in section 2.2. Thirdly, section 2.3 discusses the maritime logistics planning, relating to Hydro's Norwegian production plants. Further, we identify some challenges with the current planning process, which forms the basis for this thesis. Thereafter, the main sources of uncertainty affecting the logistics planning are described in section 2.4. Finally, the challenges with the current logistic planning process in Hydro are discussed in section 2.5. Since we study the same problem as described in Nikolaisen and Vågen (2018), there are some similarities.

2.1 The Global Aluminium Industry

Aluminium is the third most abundant element as it accounts for 8.1 weight % of the earth's crust (Lide 2005). Groover (2013) mentions several convenient properties of aluminium. Compared to steel the specific weight of aluminium is one-third, making it a lightweight metal. Its strength is easily increased in alloys. The metal has high formability and durability, thereby making it suitable for extrusion. Moreover, it has good conductivity and high resistant against corrosion, due to a natural oxide coating. Finally, the metal is also easily recyclable.

This unique combination of properties and growth in the global economy has led to a significant expansion of the aluminium industry during the past decades. Aluminium's value chain is presented in subsection 2.1.1. Thereafter, we introduce the global aluminium market in subsection 2.1.2.

2.1.1 Value Chain

To produce 1kg of aluminium, 2kg of fine, white alumina powder is required, which in turn needs 4-7 kg of raw bauxite. The first step in the chemical production is using the Bayer process to extract aluminium oxide from bauxite (Lide 2005). Thereafter, alumina is transformed to primary aluminium at production plants, called smelters, through the Hall-Héroult process. This process has a high energy intensity as an electric current is passed through alumina dissolved in an electrolytic bath, reducing it to metallic aluminium. The molten metal is periodically siphoned off and cleaned. This metal is called primary aluminium since it is not mixed with any recycled metal (American Chemical Society 2001). After the cleaning process it is mixed with various additives to create the correct alloy which is cast into different products. Examples are primary foundry alloys, sheet ingots or extrusion ingots. Thereafter, mechanical processes, such as rolling, casting or extruding, use the ingots to form the final product (Groover 2013).

Figure 2.1: Aluminium production process (Hydro 2012)

In 2017, 316 million tonnes of bauxite were produced globally. Australia and China each contributed 29% of global production, while 13% and 12% were produced by Brazil and Guinea, respectively (Hydro 2018). Because of the large reduction in weight when transforming bauxite to alumina, alumina plants are often located near the bauxite mines. Chinese alumina producers are an exception, as they rely on imports of bauxite from Guinea and Australia (Hydro 2017). Outside China, approximately 80% of alumina refining happens in close geographical proximity to the mines. For China this number is only 60% (Hydro 2018).

An overview of the geographical distribution of alumina production in 2017 is provided in Figure 2.2. The dominant market actor was China, then Oceania and thereafter South America. Comparing the production of bauxite and alumina clearly shows that China was a large net importer of bauxite, while both Oceania, primarily Australia, and Guinea were net exporters of bauxite.

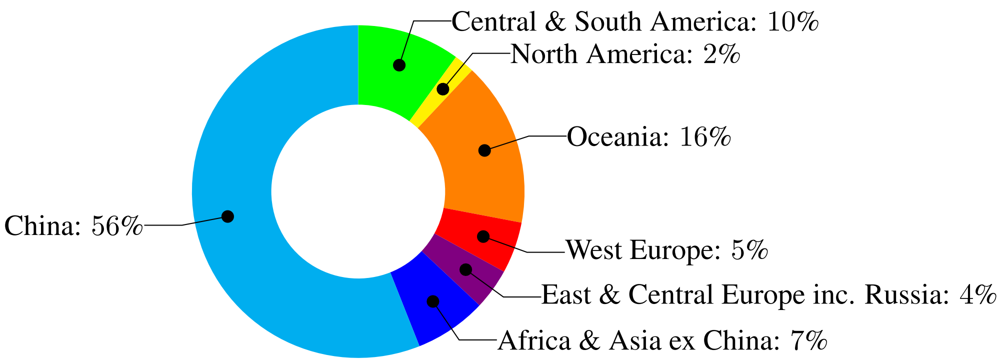

Figure 2.2: Global alumina production 2017 geographically distributed (The International Aluminium Institute 2018)

The largest cost in the production of primary aluminium is raw material, representing 35-40% of total production costs, while the cost of energy is the second largest accounting for 25-30% (Hydro 2018). Due to the high energy costs, primary aluminium smelters are usually located in countries with low energy costs. Furthermore, in order to reduce transportation costs, smelters are often located close to the consumer market. An overview of the geographical distribution of primary aluminium production globally in 2017 is shown in Figure 2.3.

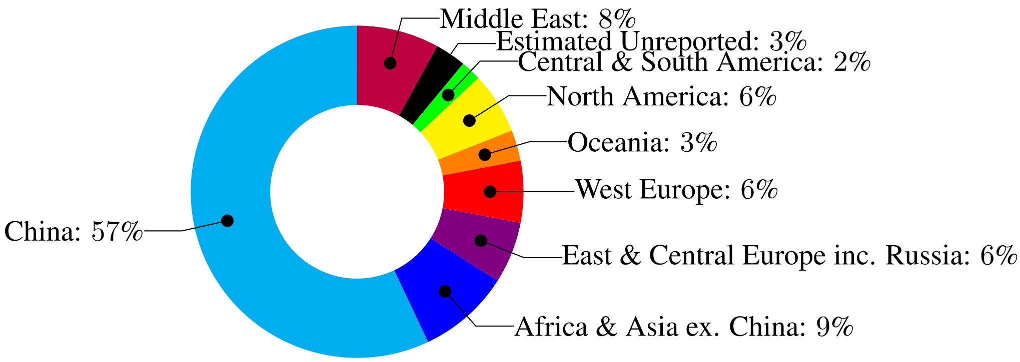

Figure 2.3: Global primary aluminium production 2017 geographically distributed (The International Aluminium Institute 2018)

By comparing Figure 2.2 and Figure 2.3, it is evident that both South America and Oceania were large net exporters of alumina, while net importers were the Middle East, North America and Europe. Due to the great distances between the smelters and the alumina supply, integrated companies operating within both alumina and primary aluminium production have a need for transport and logistics. Their production requires transportation of vast amount of bulk material over large distances.

2.1.2 Aluminium Market Conditions

In 2017 the global aluminium market, including both recycled and primary metal, was 87.5 million tonnes (Hydro 2018). Primary aluminium accounted for approximately 73% of the market (Hydro 2018). The global demand for aluminium had a 5.5% cumulative annual growth rate (CAGR) from 2000 to 2017. During this period China's CAGR was 14.7%. In the next decade the global aluminium demand's CAGR is expected to be around 3%, a more moderate rate, equal to China's expected annual growth (Hydro 2017).

An overview of the global aluminium demand in 2017, for both recycled and primary aluminium, is provided in Figure 2.4. The continent with the highest consumption was Asia at 65%. This includes China which alone contributed to 47% of the global demand. Although Figure 2.3 and Figure 2.4 cannot directly be compared due to the recycled aluminium production, it clearly shows that China was a large aluminium exporter, while North America appears to be a net importer.

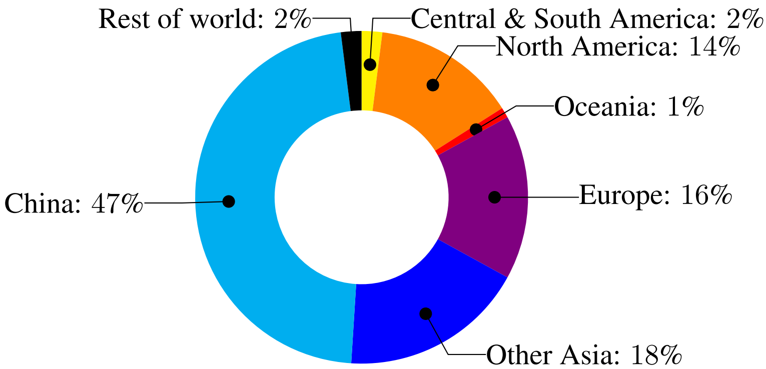

Figure 2.4: Global aluminium demand 2017 geographical divided (Hydro 2018)

Due to China's dominant market position, changes in their domestic aluminium market greatly impacts the fundamentals of the global market. In 2017 more than half of the aluminium produced globally was produced by the ten largest companies, including Hydro at ninth place. The industry is highly competitive, thereby creating incentives to increasing efficiency of operations for actors in developed countries to be able to maintain their competitive position.

2.2 Norsk Hydro ASA

The Norwegian aluminium company Hydro is fully integrated across the value chain presented in section 2.1.1. Currently, Hydro are employing approximately 35 000 workers divided between 40 countries, the majority of which are employed in Brazil, Norway, Germany and the US. In 2017, Hydro's production of primary aluminium was 2.1 million tonnes, equivalent to 3.3% of the global market, thereby making Hydro the world's ninth largest aluminium producer. Hydro's total revenue was in the same period NOK 109 billion, making it the third largest company in Norway measured by revenue. (Hydro 2018)

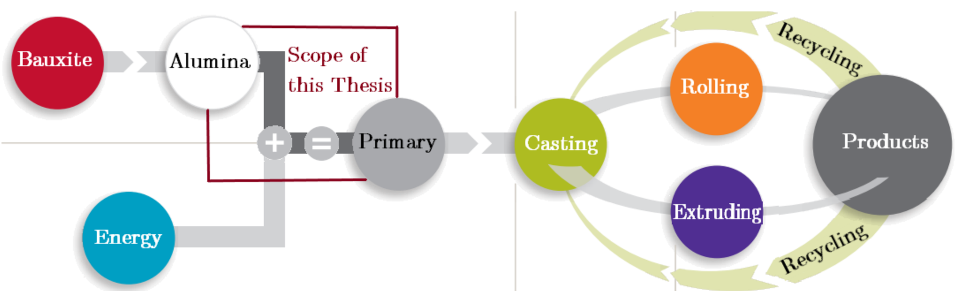

Figure 2.5: Aluminium value chain

Hydro's organizational structure reflects the value chain described in Figure 2.5. The upstream activities, Bauxite & Alumina and Primary Metal, each has its own divisions and manage a portfolio of production facilities across several countries. The downstream divisions include Rolled Products and Extruded Solutions, while the Energy division is a separate area. The subdivision for primary metal are further divided geographically with one division handling the Norwegian smelters. The Norwegian Smelters' planning division manages the logistics of transporting the alumina between the suppliers and the Norwegian smelters.

In this thesis, Hydro's activities upstream, explicitly the smelters in Norway and their supply of alumina, are considered. An overview of the upstream activities in Hydro is first provided in subsection 2.2.1. Thereafter, subsections 2.2.2 and 2.2.3 present the Norwegian smelters' alumina supply and operations in more detail, respectively.

2.2.1 Plants and Locations

Hydro's upstream section includes bauxite mining, alumina refining, electrolysis of alumina to primary aluminium and casting of the metal. An overview of Hydro's production facilities is given in Figure 2.6. Their main source of bauxite supply is the mines in Paragominas, Brazil, which produced 11.4 million tonnes bauxite in 2017. Paragominas' bauxite is refined at Alunorte, also in Brazil, which is the largest alumina refinery in the world with an annual production capacity of 6.4 million tonnes. From Alunorte, the alumina is transported to Hydro's smelters in Norway, Quatar, Canada, Slovakia, Australia and Brazil (Hydro 2018). Table 2.1 provides a full list of Hydro's smelters including their annual capacities.

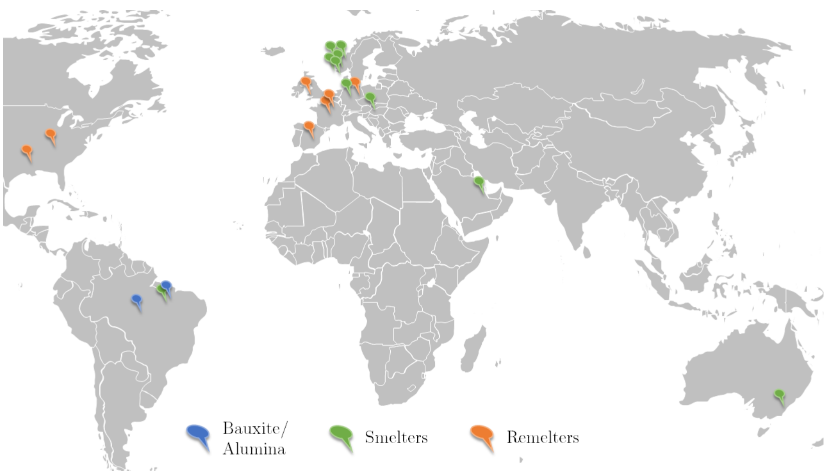

Figure 2.6: Map of Hydro's aluminium upstream production facilities (Hydro 2018)

Table 2.1: Summary of alumina plants and Hydro's proportional share of capacities (thousand tonnes) (Hydro 2018)

Plant	Country	Electrolysis capacity owned by Hydro	Casthouse capacity owned by Hydro
Karmøy	Norway	271	370
Årdal	Norway	199	230
Sunndal	Norway	407	525
Høyanger	Norway	66	120
Husnes	Norway	189	200
Slovalco	Slovakia	96	110
Tomago	Australia	74	75
Quatalum	Quatar	307	332
Alouette	Canada	122	150
Albras	Brazil	235	235

2.2.2 Alumina Supply and Sourcing

The Primary Metal division in Hydro sources alumina from internal and external suppliers through long term contracts of alumina as «Free on board». Each contract specifies an annual alumina volume, which is distributed through the period as a monthly quantity of alumina. Hydro's main external supplier is Rio Tinto Alcan (Hydro 2018).

2.2.3 Norwegian Smelter Operations

In this thesis we examine the logistical network between the alumina suppliers and Hydro's Norwegian smelters. Moreover, because the smelters outside of Norway have their own transportation and logistics planning, they are not considered in this thesis. Similarly, remelters are excluded since they use a different logistical system and their operations are mainly based on recycled aluminium. Figure 2.7 presents the geographical distribution of the suppliers and smelters considered in this thesis.

Each smelter has a port connected to its facility. Each port can only handle one ship at a time. The alumina from the ships is unloaded into silos, and the port operations can take several days. From the silos, the alumina is transported to the electrolytic cells to produce primary aluminium. Thereafter the molten aluminium is cast into ingots.

The smelters' operation is run 24 hours a day, all year. Shutdown and start-up

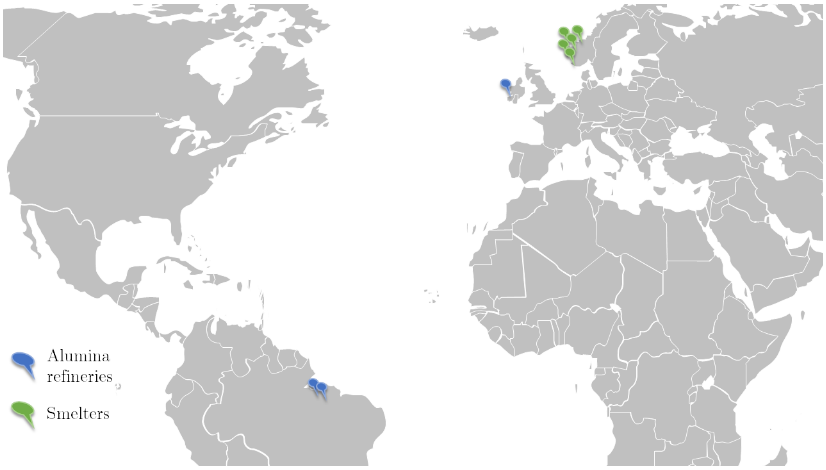

Figure 2.7: Map of Relevant Alumina Suppliers and Smelters

of the smelters is very expensive; making it critical to avoid any disruption in the electrolysis process. Therefore, it is crucial to maintain a safety stock in the silos to avoid any stock-outs. Further, the daily alumina consumption is characterized by low uncertainty with a close to constant production rate due to the continuous operations of the smelters.

2.3 Maritime Transportation Logistics

When the alumina supply contracts are signed, the planning division in Hydro negotiates long term shipping contracts with the ship owners. Thereafter, a team in Oslo manages both the operational and tactical logistic planning of transporting the alumina to the Norwegian smelters. In this section, the shipping contracts are first discussed in subsection 2.3.1. Then, today's logistic planning process is described in subsection 2.3.2.

2.3.1 Shipping Contracts

Hydro is engaged in several long-term shipping contracts with various shipping companies offering different bulk carriers. In the contracts an interval for the total number of shipments per year is specified. They also include the characteristics of each ship class such as speed, size, transportation cost and capacity. The cost of transportation depends on the ship class. It is given as a cost per route and quantity transported. When splitting loads between unloading ports the shipping company charges an additional freight cost.

Between three to four weeks in advance of a shipment, Hydro is required to notice the shipping company. This is done by making a call-off on the shipping contract where the ship's routing, the amount to load and unload at each port and time window for loading is specified. Hydro is free to request any of the ship classes for the trip. Thereafter, the shipping company nominates a specific ship belonging to the requested class. To finalize the agreement both the loading port and Hydro needs to approve the ship assigned.

2.3.2 Logistic Planning Today

Tactical and operational planning is in practice done interchangeably. However, to be able to clarify the timeline of the planning process we will in this subsection highlight distinctions between the two planning levels.

Operational planning involves making call-offs and manage unforeseen events occurring between the unloading at the aluminium plants and up to three to four weeks in advance. After making a call-off, the loading quantities and time window in the given loading port are not open to amendment. Hence, the planning division has limited opportunity for changing the plan regarding fleet mix and loading decisions. However, the decisions regarding unloading ports are not locked at such an early stage. Hence, ships may be rerouted, and the unloading quantities may be changed up to a few days before arrival.

Tactical planning involves determining the delivery plan and shipping schedule for the next one to six months. During this planning, a planned shipment is assigned a ship class, not a specific ship, since the exact ship is not known ahead of the call-off.

Making a call-off on the shipping contract, connects a specific trip with a specific ship. Further, a time window of one week is selected by the planning division for when the loading operation can begin at the loading port. The exact timing of loading is uncertain and can be any time within the time window. Furthermore, due to weather conditions, travelling times are uncertain as well.

Currently, the planning process for maritime logistics to the Norwegian smelters are mostly done manually in Excel. It is based on a deterministic view, assuming that the ships will begin loading in the middle of the time window. The planning division performs some worse case scenarios by checking the planned schedule's robustness. This process consists of separately considering how the result is influenced by only late arrivals, as well as only early arrivals.

2.4 Sources of Uncertainty

There are several sources of uncertainty that affect Hydro Norway's alumina supply. In this thesis, two main sources are considered. Firstly, the actual time at which a ship begins loading at the loading port can be any time within the one-week time window that is agreed with the port. Hydro has no way of affecting when loading starts within this window. Secondly, weather conditions can impact the sailing time up to ± 2 days. Therefore, the greatest source of uncertainty is the start time of loading at the alumina plants. As an effect of these uncertainties, the actual arrival time of a ship in Norway can be significantly different from the planned arrival time. In order to avoid stock outs, this uncertainty must be considered in the tactical planning problem.

2.5 Challenges with Today's Process for Logistic Planning

Today, the team planning maritime transport has little decision support in the planning process, meaning that most of the planning is done with a deterministic approach in Excel. For tactical planning, considering the full backlog of possible routes and ships for the upcoming one to six months is challenging due to the size of the problem. The situation is further complicated by the uncertainty in when the ships begin loading at the alumina plants and the uncertainty in travelling time. Finally, the possibility to re-route ships in the operational planning phase as well as the possibility to split the cargoes create an additional layer of complexity. Hence, today's planning process' main challenge is the absence of decision support when attempting to optimize a large logistical system where stock-outs at smelters have great consequences.

The focus of this Master's thesis will be on the tactical planning including some operational aspects. It considers the maritime transportation of alumina between the alumina plants and the primary aluminium smelters in Norway. The main alumina plant is Alunorte in Brazil. Further, we consider supply from Rio Tinto Alcan, Hydro's main external supplier (Hydro 2018), and Aughinish in Ireland, Europe's largest supplier. The Norwegian smelters have a predictable, fixed production rate, and we treat the smelters' consumption as known and constant.

Chapter 3

Problem Description

In this chapter we present the multi-product maritime inventory routing problem (MIRP) faced by an international aluminium producer. The fleet of ships is heterogeneous and splitting loads between the unloading ports is allowed. Further, the problem has a tactical planning horizon. Two key sources of uncertainty are introduced to make the problem more realistic. The output generated from the MIRP is a set of schedules and routes for the fleet of ships including the quantity of product collected at each port visit. The deterministic version is similar to the problem described by Nikolaisen and Vågen (2018).

In section 3.1 we describe the different elements of the models, while section 3.2 introduces the uncertain elements of the problem. Lastly, the objective and decisions of the models are discussed in section 3.3.

3.1 Elements of the Model

This section presents the various elements defining the model. The costs and operations at the loading- and unloading ports are discussed in subsections 3.1.1 and 3.1.2, respectively. Further, the characteristics of the fleet of ships are described in subsection 3.1.3. Lastly, subsection 3.1.4 introduces the different product qualities.

3.1.1 Loading Ports

Each month, the total quantity loaded at a loading port must be below a certain maximum and above a given minimum, in accordance with the supply contracts. As specified in the supply contracts, alumina is bought "Free on board". Therefore, inventory management at the loading ports is not considered. Each port has one berth available for loading. The time window to begin operation is decided for the ships.

For each ship, the setup time for loading is fixed in each loading port. The loading rate, in quantity per hour, for a given ship in a given port is fixed and determines the loading time. Further, we assume that a ship is fully loaded when it leaves a loading port. A port fee is charged for each visit.

3.1.2 Unloading Ports

Similar to the loading ports, one berth is available for unloading at every unloading port. Further, each unloading port has a port fee, a set up time for unloading and a given unloading rate, in quantity per hour.

The product is stored in silos at the unloading ports. The silos have a storage capacity and a safety stock requirement. As a simplification the silos are considered in aggregate, providing accumulated limits for both the upper and lower inventory level at each unloading port. Stock-outs are not tolerated, since the cost associated with a stop in production is high. Due to stable production rates at each plant, the consumption is treated as constant and known through the period.

3.1.3 Ships

Although the shipping companies offers a heterogeneous fleet of ships, we assume the fleet can be represented by a set of heterogeneous ship classes, each consisting of homogeneous ships. Every ship in a class has the same velocity, capacity, cost structure and size. The daily transportation costs include the cost for renting and operating the ship. When modelling with daily transportation costs there is no need for additional costs for splitting loads, because the cost is captured by the extra operating days and port fees.

As discussed in subsection 2.3.1, there are long term contracts with several shipping companies. As a simplification, we accumulate all the contracts for the entire period to provide an upper and lower bound for the total number of shipments. Further, we assume that all ship classes are available for charter at all times.

3.1.4 Product Qualities

The quality of the raw material varies depending on the loading port, and the preferences for quality varies depending on unloading port. To accommodate this, a compatibility check is introduced to disallow certain combinations of quality and unloading ports.

3.2 Sources of Uncertainty Considered in this Thesis

In order to make the problem more realistic, uncertainty is introduced. There are two sources of uncertainty that will be considered in this thesis. Firstly, the time that a ship is allowed to start loading in a loading port is stochastic in nature. Each ship is given a time window of one week, but the time at which it will start loading in this time window is uncertain. Secondly, the travel time from the loading port to the unloading ports is uncertain, due to weather conditions, and can vary with ± 2 days. Hence, the primary source of uncertainty in the system is the start time of loading which can vary within a week.

3.3 Objective and Decisions of the Model

The objective of this thesis is to create a shipping schedule over a tactical time horizon determining routing, ship class, timing of trips and quantity of material loaded and unloaded at ports. This schedule should perform well under the uncertainties discussed in section 3.2 and minimize expected costs. Nikolaisen and Vågen (2018) conclude that it is efficient to minimize the cost of transportation, port fees, operations and waiting and thereafter consider inventory costs separately in a simple post processing heuristic. Therefore, we exclude the inventory cost for this problem, since including them complicates the problem considerably (Nikolaisen and Vågen 2018). Thus, the expecting costs we minimize consist of costs for transportation, port fees, operations and waiting.

We assume that the initial position of a ship is an artificial origin. Some ships are assumed to be in transit at the origin. These ships have an initial load and deterministic travel times to all the unloading ports. If a ship is empty at the beginning of the time horizon, the travel times from the origin to all the loading ports are zero days. As described above, the start times of operations in the loading ports are uncertain as well as the travel times between the loading and unloading ports. Further, we assume that all cargo on a ship is unloaded before the end of the time horizon.

The return voyage is not taken into consideration in this thesis, due to the fact that each ship is only charted for a specific trip between loading and unloading ports. Therefore, we assume that during the planning horizon a ship makes at most one trip. Moreover, we assume that the supply and shipping contracts always are sufficient to allow feasible solutions. Hence, the option to charter a ship from the spot market is not included.

Each ship can only visit one loading port but it is allowed to split the cargo between multiple unloading ports. Further, we assume that there is an upper limit for how many unloading ports a ship can deliver to. Finally, there is some flexibility in re-routing the ships after they have left the loading ports. A ship can be redirected to a different set of unloading ports than planned up until 3 days before the planned arrival and the unloading quantities can also change.

Chapter 4

Literature Review

The purpose of this chapter is to provide an overview of the literature on how uncertainty can be handled in Maritime Inventory Routing Problems (MIRP). The intention is to present a selection of important papers relevant for this thesis, rather than giving a comprehensive review of all relevant literature for MIRPs and handling of uncertainty.

Firstly, in section 4.1 we present the same classification as Stålhane (2013) of Maritime Transportation Problems categorized along two axes. Then, a brief presentation of the MIRP is provided in 4.2. Lastly, section 4.3 discusses the different solutions methods commonly used for dealing with uncertainty in the MIRP literature with examples.

4.1 Classification of Maritime Transportation Problems

Maritime transportation problems are often categorized along two axes in the literature; the planning level and the transportation mode (Stålhane 2013). They are described in subsections 4.1.1 and 4.1.2, respectively. Then we categorize this thesis problem according to the axes in subsection 4.1.3.

4.1.1 Planning Level

The planning level is divided into three sublevels; strategic planning on the top, tactical planning beneath and then operational planning. The levels are differentiated by several factors, such as the length of the time horizon and the type of decision. Anthony (1965) was the first to introduce the taxonomy, which has later been applied by Hax and Meal (1973) to production planning and by Christiansen et al. (2007) to maritime transportation problems. Decisions taken on one planning level affects the other levels. Especially, decisions made on the upper levels usually impose some of the boundary conditions for the levels beneath (Christiansen et al. 2007).

The top level is known as strategic planning. In maritime transportation it usually has a time horizon from 5 up to 20 years (Stålhane 2013). Further, it typically concerns decisions for the ship supply (Christiansen et al. 2007). Examples are ship design, market selections, maritime logistic system design, fleet size and mix and transportation system. Tactical planning, on the other hand, has a time horizon between some weeks and up to 1.5 years (Stålhane 2013). The focus for tactical decisions are typically ship routing and scheduling or fleet deployment (Christiansen et al. 2007). The lowest level is categorized as operational planning (Bitran and Tirupati 1993) with a time horizon of a few hours up to a few weeks in maritime transportation (Stålhane 2013). Operational decisions are typically concerned with re-routing or the sequence of port operation for ships (Christiansen et al. 2007).

4.1.2 Transportation Mode

Christiansen et al. (2007) and Stålhane (2013) distinguish between three different types of transportation modes; liner, tramp and industrial shipping. When compared to land base transportation mode, liner shipping is best represented by a bus service; each ship earns profit from transporting cargos between the different ports in a given route according to a fixed schedule. Most of the liner shipping companies transport containers. The objective of liner shipping is to maximize the profit earned from servicing the cargoes.

Tramp shipping also has the same objective as liner, but the operation of each ship is more similar to a taxi service without a fixed schedule. Each shipping company tries to maximize their earnings from pickup and deliver the cargoes. Operators in tramp shipping usually control dry bulk carriers or tankers.

In industrial shipping the ships usually transport high volume of dry bulk or liquids, such as: solid raw materials, chemicals, and oil. The objective is to minimize total cost of the services. Typically, the cargo owner also controls the shipping operation, thereby increasing the scope of the problem by including a greater part of the supply chain. The routing and scheduling aspect could for example be combined with inventory management and/or sales or production. Maritime inventory routing problems, hereby abbreviated to MIRP, is an example of combining the scheduling and routing aspect with inventory management.

4.1.3 Placing the Thesis' Problem along the Two Axes

The problem in this thesis is an industrial shipping problem since the company considered owns the goods transported and controls the inventory in the unloading ports. Further, the planning horizon is tactical with a time horizon of between one and a half to three months. There are also some operational planning aspects because rerouting of ships based on realized uncertainty is considered. However, the main objective is to create a good shipping schedule over a tactical planning horizon.

4.2 Maritime Inventory Routing Problems

MIRPs combine inventory management with maritime routing and scheduling problems (Christiansen et al. 2013). However, the MIRP does not have a clear definition, but is rather a collective term. Therefore, the variety of aspects that can arise and assumptions that are made is large. According to Andersson et al. (2010) this is the reason why every new paper published in the MIRP literature often presents a new version of the MIRP.

This section provides a brief introduction to MIRPs. For more extensive surveys of the topic, we refer to Christiansen and Fagerholt (2009, 2014), Christiansen et al. (2013) and Papageorgiou et al. (2014). First, some of the problem characteristics that we consider relevant to this thesis are presented in 4.2.1. Then, two key modeling choices for MIRPs are discussed in 4.2.2.

4.2.1 MIRP Problem Characteristics

According to Andersson et al. (2010) MIRPs can vary along different dimensions. In Table 4.1 we have restated a non-exhaustive selection of the characteristics to highlight some of the variety, and how difficult they are to solve.

Table 4.1: A variety of MIRP dimensions

MIRP characteristic description	Simpler	More complex
Rate of consumption/production	Constant	Optimized
Nr of products	Single	Multiple
Split pick-ups / deliveries	Forbidden	Allowed
Inventory management at ports	One/few	All
Port operational time as a function of quantity (un)loaded	No, constant	Yes

Christiansen (1999) is one of the first MIRP papers published and also one of the most cited articles in the MIRP literature. Christiansen's MIRP allows for split pick-up and deliveries, has incorporated variable operating time as well as visit time windows. Therefore, it is quite a rich MIRP. The model Christiansen (1999) developed provides the basis for several papers later published (Agra et al. 2014), such as Al-Khayyal and Hwang (2007), Siswanto et al. (2011) and Agra et al. (2013b). The ones mentioned all focus on liquid bulk shipping. Further, they extend the model to handle multiple products.

Agra et al. (2016) describe another MIRP characteristic. They distinguish between the short sea inventory routing problem (SSIRP) and the deep sea inventory routing problem (DSIRP). DSIRPs is intercontinental, usually with longer travelling times and therefore longer time horizons compared to SSIRPs (Hemmati et al. 2016). Examples of a DSIRP are Rakke et al. and Rakke et al. (2015) both considering a tactical liquefied natural gas (LNG) MIRP to construct an annual delivery program (ADP). They do not allow split deliveries. Further, they assume constant port time and a production rate dependent on time.

Furthermore, SSIRPs' ratio between traveling time and port operating time is higher than the ratio for DSIRPs. Hence, models considering SSIRPs often need to consider operational time in ports as a function of quantity (un)loaded, whereas models considering DSIRP usually consider operational time in ports as fixed (Hemmati et al. 2016). Examples of SSIRPs are Christiansen (1999) and the mentioned papers working with a refined version of Christiansen's model, as well as Agra et al. (2013a) and Agra et al. (2018). Both consider a single product MIRP allowing for split deliveries and operational time as a function of quantity (un)loaded. While the former considers a tactical planning level, the latter has an operational planning level.

4.2.2 Modelling Choices for MIRPs

Apart from different classifications of MIRPs according to their problem description, characteristics of the mathematical programming formulations also separate them. According to Papageorgiou et al. (2014) at least two dimensions classify the formulations.

Firstly, the modelling of the time can be either discrete or continuous. In a continuous time model, time is used as a continuous decision variable, such as visiting time t_{imv} indicating which port i has its visit number m by ship v. On the other hand, a discrete time model treats time as a discrete index for the variables and parameters, such as the binary variable x_{ivt} indicating if port i is visited by ship v in time t, thereby constraining events to only take place at fixed points in time. Agra et al. (2016) and Nikolaisen and Vågen (2018) have compared the two types. Further, both concluded that discrete time formulation has both a larger size, compared by number of variables and constraints, and requires a longer runtime to reach optimal solution. Further, this difference increases when increasing the length of the time horizon.

Secondly, the formulation can be either path-based or arc-based. The decision variables in the former represent the entire port visit sequence for each ship, while in the latter formulation they model each ship's movement between port pair (Christiansen et al. 2013, Papageorgiou et al. 2014). Further, arc formulation can be devided into arc-load and arc-load flow. While both models track information regarding the ships' load during each port visit, the latter also has variables describing the cargo quantity on each ship traversing an arc. Several articles such as (Agra et al. 2013b) have shown that the latter is more efficient. Hence, we will use a continuous time arc-load flow formulation in this thesis.

4.3 Uncertainty Handling Methods in MIRP Literature

Even though the shipping industry is greatly influenced by uncertainty, most of the MIRP literature considers deterministic problems (Agra et al. 2018). When uncertainty is considered in MIRPs it often occurs in demand/consumption of a product or in the time parameters such as travelling times, port operational times or arrival times in ports. In this thesis we will focus on the latter. Uncertainty in time parameters is usually due to unknown weather conditions or delays occurring due to contracts with external companies, for example port strikes or late deliveries (Zhang et al. 2018).

In this section we discuss different ways to handle uncertainty in the MIRP-literature. Firstly subsection 4.3.1 presents how some MIRP articles that try to handle uncertainty by using a modified deterministic model. Thereafter, robust optimization and the use of it in MIRP is presented in subsection 4.3.2. Subsection 4.3.3 considers the use of stochastic optimization in MIRP. Finally, subsection 4.3.4 presents the combination of simulation and optimization and provides some examples of MIRP papers implementing such a framework.

4.3.1 Policy Implementations in a Deterministic Model

Several MIRP articles have tried to handle uncertainty by modifying a deterministic model using different strategies. Examples of strategies include adding soft penalties in the objective function or changing the modelling parameters to introduce system slack. These methods were used in the MIRP literature to better handle uncertainty earlier than both robust and stochastic optimization, as the first robust optimization MIRP paper was Agra et al. (2012), while the first stochastic optimization MIRP paper was Agra et al. (2015).

Soft penalty functions can be added to make certain inventory levels or arrival times unfavourable by incorporating minor modifications to the deterministic model. Such is done by Christiansen and Nygreen (2005) to handle uncertain travel and port times. They include soft inventory levels, penalizing inventory outside the soft boundaries. Furthermore, they transform the levels into soft time windows. The latter is also used by Christiansen and Fagerholt (2002) with the purpose of designing ship schedules which try to prevent ships staying idle during the weekends.

When changing the input parameters, the same deterministic mathematical model can be used to manage uncertainty. Examples for MIRP could be to introduce safety stock levels or increase the time parameters such that delays can be absorbed. Therefore, the solutions are more robust to delays. Such strategies are both tested and evaluated in several papers, for example in Halvorsen-Weare et al. (2016) and Fischer et al. (2016). Both papers explore several strategies, such as increasing the travel time between ports. Furthermore, the effects are evaluated by simulation. Halvorsen-Weare et al. (2016) draw the conclusion that the robustness strategies resulted in solutions with overall lower expected costs than the basic approach. Due to the use of simulation, we will further discuss both Halvorsen-Weare et al. (2016) and Fischer et al. (2016) in subsection 4.3.4.

4.3.2 Robust Optimization

Robust optimization as first introduced by Soyster (1973) tries to protect solutions against uncertainty, by making them feasible for all realizations of the random events, and not allowing recourse decisions. To ensure feasibility for all realizations, only the border of the uncertainty polytope is required, and not the distributions of the uncertain parameters (Soyster 1973). Therefore, the problem may be as simple to solve as the original problem. Soyster's robust optimization is also known as worst-case analysis, since the solution is feasible for all realizations, a highly conservative approach.

Bertsimas and Sim (2004) develop a new robust approach considering the trade-off between robustness and conservatism of a solution. This is done by introducing the parameter Γ_i to adjust the level of the robustness of the solution. Their approach protects against all realizations where up to $\lfloor \Gamma_i \rfloor$ of the uncertain coefficients are allowed to change, which is often referenced as the uncertainty budget (Bertsimas and Sim 2004). This budget reflects the attitude towards uncertainty to the decision maker (Diz et al. 2018). Note that if Γ_i is equal to the number of uncertain coefficients the formulation are the same as Soyster (1973). The formulation of Bertsimas and Sim has the same complexity as the deterministic optimization problem and does not require the probability distribution of the uncertain parameters.

Adjustable robust optimization developed by Ben-Tal et al. (2004) is another interesting robust technique worth mentioning. Adjustable robust optimization is a more flexible class of robust programs compared to Bertsimas and Sim (2004). They introduce two stages of decisions where a subset of variables can change after the uncertainty is revealed. By allowing some variables to change, they incorporate flexibility in the problem (Ben-Tal et al. 2004).

Agra et al. (2012) are the first to present a general approach to the robust vehicle routing problem with time windows and uncertain travel times. Their formulation is motivated by maritime transportation and applies the concept of robust programming approach presented by Bertsimas and Sim (2004). They add complexity by letting the travel times belong to an uncertainty polytope, which creates a problem that is harder to solve than the deterministic counterpart, but creates less conservative robust solutions. As a consequence the number of instances they are able to solve is limited (Agra et al. 2012).

Diz et al. (2018) also adopt the approach proposed by Bertsimas and Sim (2004). Further, they use a matheuristic to solve the robust optimization model for each level of Γ. Thereafter, to check the probability of infeasibility, they use a simulation process which selects the uncertain parameters based on historical data. They conclude that robust optimization improves the decision quality considerably by reducing the risk of infeasibility for a limited increase in operational costs (Diz et al. 2018).

The limited size of the instances Agra et al. (2012) are able to solve is the motivation for Agra et al. (2013c), presenting two new formulations for the same problem. The first formulation is based on resource inequalities, where canonical cuts substitute robust constraints. The second formulation is based on path inequalities and uses adjustable robust optimization as described by Ben-Tal et al. (2004). Agra et al. (2013c) use decomposition algorithms to solve larger instances than Agra et al. (2012) with both formulations.

Agra et al. (2018) build an adjustable robust program as introduced by Ben-Tal et al. (2004). The model uses a two-stage decomposition procedure considering a master problem and a subproblem, checking if a solution is feasible for a small subset of scenarios and then all scenarios, respecively. Agra et al. (2018) use

 26

instances with up to 6 ports, 5 ships and a maximum of $\Gamma = 4$ links that can suffer delays. They show that increasing Γ results in a great increase in computational time since both runtime and number of iterations tend to increase, even after the implementation of several improvement strategies.

4.3.3 Stochastic Optimization

In stochastic optimization, the information structure of the problem plays a key role. During the time horizon, realizations of random events will occur. The first decisions made, called first-stage decisions, are taken without full information regarding the events, considered as the robust variables, thereby generating a solution independent of the outcome. Later on, we receive more information on the realization and corrective actions, called the recourse decisions, are taken dependent on the outcome of the random event (Birge and Louveaux 2011). The information structure can be two-stage, where all uncertainty is revealed at once, or multistage where there are multiple sets of recourse decisions.

To be able to solve stochastic optimization, a probability distribution for the random variables is needed. Further, the stage structure of the problem needs to be clearly defined, regardless of the number of stages (Birge and Louveaux 2011). Therefore, the number of scenarios to consider quickly becomes very large. This is particularly true for multistage stochastic optimization and often leads to large models (Birge and Louveaux 2011). Because of the complexity of deterministic MIRPs, stochastic optimization MIRP models have received limited attention. To the best of our knowledge, three articles consider two-stage stochastic optimization and no articles have attempted to use multistage stochastic optimization.

The first MIRP paper to use stochastic programming as a solution method was Agra et al. (2015). They consider random sailing and port times with relatively complete recourse, implying that all first stage decisions will result in feasibility in stage two. Further, due to the large number of scenarios they use the L-shaped method and sample average approximation (SAA) to reduce the solution time (Birge and Louveaux 2011).

Zhang et al. (2018) also formulate a two stage stochastic program for a MIRP to develop an annual delivery plan for shipping LNG. They treat the length and placement of time windows as decision variables. They acknowledge that the number of scenarios is too great to solve the problem within a reasonable amount of time. Therefore, they propose a two-phase solution approach, where the first phase is solving a robust MIRP where the placement and length of the time windows are decision variables, while the second phase considers various types of disruption which may affect ship availability and travel time.

4.3.4 Simulation and Optimization

As discussed, MIRPs are challenging to solve when uncertainty is involved. While optimization generates solutions that are presumed to be optimal under the given settings, simulation evaluate solutions without taking any decisions (Fu 2002). When the problem under uncertainty is hard to optimize, the different characteristics suggest that the methods can be good complements. Fu (2002) is a good introduction to the simulation and optimization topic, while Figueira and Almada-Lobo (2014) try to provide an overview of the full spectrum of approaches for simulation-optimization, including a comprehensive taxonomy.

Figueira and Almada-Lobo (2014) divide simulation-optimization research into three major streams; "solution evaluation", "solution generation" and "analytical model enhancement". The former corresponds to the "simulation for optimization" described by Fu (2002), where simulation is used to evaluate the optimization solution. This has previously been the main focus in simulation literature, while the two latter have been the main focus in optimization literature. They combine simulations with an optimization model and are known as "hybrid simulation-analytic models/modelling" (Figueira and Almada-Lobo 2014). Because optimization is the main topic of this thesis, we will hereafter focus on the "hybrid simulation-analytic" streams.

Using Simulation for Solution Generation

The main purpose of the simulation in a "solution generation" approach is to contribute to the generation of a solution by using simulation to compute some of the variables in addition to the ones decided initially by the optimization model (Figueira and Almada-Lobo 2014). Hence, the simulation is ran once and evaluation of different solutions' performance under uncertainty is not the main purpose (Figueira and Almada-Lobo 2014). This could be done as well, by running the framework several times, which is often a time consuming process.

One example of "solution generation" approach with evaluation in MIRP literature is Halvorsen-Weare et al. (2016), with the aim of creating more robust routing and scheduling solutions for LNG ships. They implement a simulation model with a recourse optimization procedure to evaluate the output from an initial optimization model with a variation of policy strengthening strategies. The recourse optimization model is called when certain events occur, leading to re-routing of the ships in the remaining time period. This type of simulation where recourse decisions are allowed is known as "solution completition by simulation" (Figueira and Almada-Lobo 2014). In order to reduce runtime, Halvorsen-Weare et al. (2016) use a re-routing model without any strategies.

Halvorsen-Weare et al. (2016) need to run the simulation framework multiple times to compare the performance of the policies. This substantiate that using "solution generation" to evaluate different policies where the simulation feedback is important can be quite time consuming, as the model must be run several times (Figueira and Almada-Lobo 2014). Note that the article by Zhang et al. (2018) also uses simulation to evaluate solutions, by creating a simulator for disruptions with the possibility of creating recovery solutions.

Fischer et al. (2016) also evaluate a variation of strategies for disruption management by designing a simulation framework with a recourse model. They study a tactical, maritime fleet deployment problem in liner shipping. Firstly, an initial solution is created by a rolling horizon heuristic (RHH) and fed into the simulation framework. If an event occurs requiring replanning, they call a recourse model such as Halvorsen-Weare et al. (2016). However, instead of having a new model, they uses the RHH from the trigger event to the end of the time horizon (Fischer et al. 2016).

The paper by Dong et al. (2018) is an example of a MIRP "solution generation" using iterative optimization-based simulation by creating a rolling horizon optimization framework for a discrete-time mixed-integer formulation. Dong et al. (2018) simulate each period and check if the optimization solution is valid. If it is not valid, a reoptimization program is called. The process is similar to a RHH as in Fischer et al. (2016), but instead of optimizing only initially, it calls the optimization problem to generate a new solution for the rest of the horizon.

Using Simulation for Optimization Model Enhancement

"Analytical model enhancement" is another approach, where the simulation model is used to enhance the optimization model. This is done by using the simulation model's output as feedback into the optimization model. The feedback is either used to extend the optimization model or refine its input parameters. As to our knowledge, the number of MIRP papers using this approach is scarce, hence we present some papers from other problem fields as well.

Nolan and Sovereign (1972) were the first to develop an iterative solution approach between simulation and optimization. It is problem specific, involving allocation of resources on an aggregated level by an optimization model and revising the production estimates in simulation. Other papers like Karabakal et al. (2000), Lee et al. (2002) and Ko et al. (2006) also develop a problem specific hybridized framework where simulation enhances the optimization model by producing estimates on uncertain parameters, which is fed into the next iteration of the optimization model. The same accounts for De Angelis et al. (2003), but instead of interchanging constraints they rather enhance the optimization model by updating the objective function.

All of the latter six articles mentioned develop problem specific iterative solution procedures where problem-specific parameters are interchanged. Acar et al. (2009) are the first to develop a general solution methodology which obtains an estimated global optimum for combinatorial optimization problems incorporating uncertainty. Figueira and Almada-Lobo (2014) characterize the methodology as a “recursive optimization-simulation approach”, a version of “analytical model enhancement”. It is described as running a deterministic optimization model and a simulation model alternately. The latter computes performance measures of the specific solution, which is fed into the optimization in later iterations. The iterations terminate when meeting a stopping criterion, such as convergence of the objective, parameters or solution. Acar et al. (2009) is geared to solution spaces with discrete characteristics.

Holm and Medbøen (2017) use a similar approach as Acar et al. (2009) for a maritime routing problem in liner shipping with uncertain weather conditions. Their solution framework feeds solutions from an optimization model into a simulation model. The latter simulates the performance of the solutions under uncertainty as well as applying replanning and recovery actions when needed. After the simulation, a penalty cost for each solution is feed into the optimization model. Then the optimization model is reoptimized until a solution already simulated is picked as optimal.

30

Chapter 5

Solution Framework

In this chapter we present the combined optimization and simulation framework to handle the uncertainty in the system described in chapter 3. The framework is based on the iterative approach proposed by Acar et al. (2009). First, the motivation behind our choice of solution method is explained in section 5.1. Then, the stage 1 optimization model and stage 2 simulation procedure are described in detail in sections 5.3 and 5.4, respectively. Finally, techniques for providing feedback to stage 1 after simulating stage 2 are discussed in section 5.5.

5.1 Motivation for Design of Framework

As discussed in section 3.3, the objective of this Master's thesis is to generate shipping schedules that are expected to perform well under uncertainty in travel time and start time of loading. For the aluminium producer Hydro, decisions relating to which ships to charter and when to load at the loading ports are difficult to change on short notice. However, once the ship is chartered and loaded, Hydro has some flexibility to alter its routing and unloading decisions. Therefore, we want to create a planned shipping schedule that is robust with respect to the fleet mix and the loading decisions. Furthermore, we also want to allow some flexibility in the routing and unloading to adapt the schedule after the uncertainty is revealed.

Thus, we have a problem similar to stochastic optimization where some first stage decisions must be taken before the uncertainty is known, while some recourse decisions are postponed until after the uncertainty is revealed. However, the literature review indicates that it is challenging to solve stochastic models within a reasonable runtime for a problem of our size and complexity (Agra et al. 2015). Furthermore, in order to use stochastic programming, a stage structure needs to be

32

clearly defined. The information structure of the problem discussed in this thesis is multistage in nature where information about the departure and travel times for each ship is revealed at times that are themselves random.

Therefore, we propose a combined optimization and simulation framework based on the ideas presented by Acar et al. (2009). This approach allows us to incorporate re-routing decisions as well as multistage information structure. As a simplification we first model the problem with a two-stage information structure where all uncertainty is revealed between the stages. However, we extend the problem to imitate a multistage structure in chapter 6.

5.2 Solution Framework

The combined framework is illustrated in Figure 5.1 and is based on the generalized framework presented in Acar et al. (2009). A deterministic model is first used to create a planned schedule for the entire time horizon in stage 1. Then, the performance of this potential solution is evaluated using simulation with a re-routing model in stage 2. After stage 2, the estimated cost of uncertainty is associated with the potential solution from stage 1. Thus, when the stage 1 model is run in the next iteration, a different solution will likely be chosen because the original solution has an extra penalty cost equal to the estimated cost of uncertainty. This process is repeated until the stage 1 model chooses a solution that it has already chosen before.

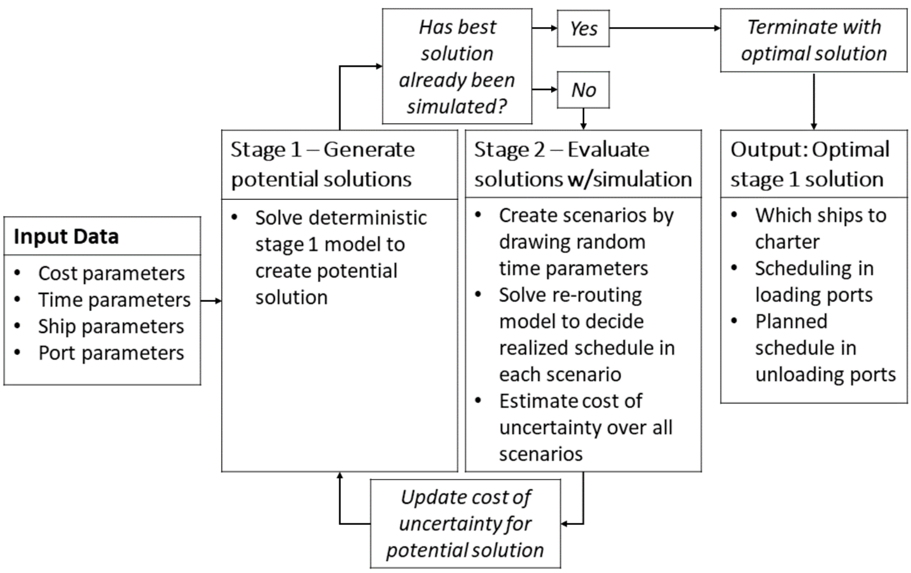

Figure 5.1: Flow chart of the optimization-simulation framework

In stage 2, the actual times of loading and travel times are drawn randomly to create the different scenarios. For each scenario, a re-routing model is used to determine the optimal routing of the ships after uncertain parameters has been known. Some of the routing and unloading decisions are allowed to change compared to the planned schedule to introduce flexibility to the problem. The cost of uncertainty for a potential solution from stage 1 is estimated as the average difference between the planned deterministic cost and the realized cost over $|S|$ simulations.

Furthermore, the re-routing problem will be infeasible for some realizations of the uncertain parameters. Solutions can only become infeasible due to stock-outs in the unloading ports. In these instances, the cost of uncertainty is calculated by adding a stock-out cost to the planned cost of the solution. The value of the stock-out cost controls how conservative the optimal stage 1 solution will be. A high stock-out cost will favour stage 1 solutions that always remain feasible in the stage 2 evaluation.

We believe this framework has three favourable features that make it well suited to our problem. Firstly, we capture the flexibility of the problem by allowing certain recourse decisions in stage 2. This represents an advantage compared to Bertsimas and Sim's robust optimization. Secondly, we avoid including the scenario space in the optimization model by sampling the uncertain parameters through simulation. Because of the complexity of the deterministic problem, a stochastic model with scenario representation would likely be very difficult to solve. Finally, the stage structure allows us to model the information structure as either two-stage or multistage. A multistage representation would, in our opinion, be very difficult to implement with both robust optimization and stochastic programming.

5.3 Stage 1 – Solution Generation

The purpose of stage 1 is to generate potential solutions to be evaluated in stage 2. Stage 1 involves creating a planned shipping schedule for the entire planning horizon using a deterministic model with expected travel times. This problem is similar to the problem considered in Nikolaisen and Vågen (2018) and our mathematical model is a refinement of this work. Some changes have been made to the network structure to facilitate the stage structure of our solution framework. Further, some new valid inequalities are also introduced.

This section begins with a discussion of the network structure of stage 1 in subsection 5.3.1. Then, we elaborate on the flexibility of the system in 5.3.2, explaining which decisions that are final after stage 1 and which can be altered in stage 2. Finally, the mathematical model for stage 1 is presented in 5.3.3.

5.3.1 Changes to the Network Structure

In the original stage 1 problem there is a set of loading and unloading ports, with a network structure like the one illustrated in Figure 5.2. In this thesis, all the unloading ports are located in the same region, namely the west coast of Norway. Therefore, all ships need to pass through the North Sea before reaching their destination. We utilize this fact by creating a transit point in the North Sea that all the ships must travel through. This modification to the network, illustrated in Figure 5.3, is essential to the simulation framework.

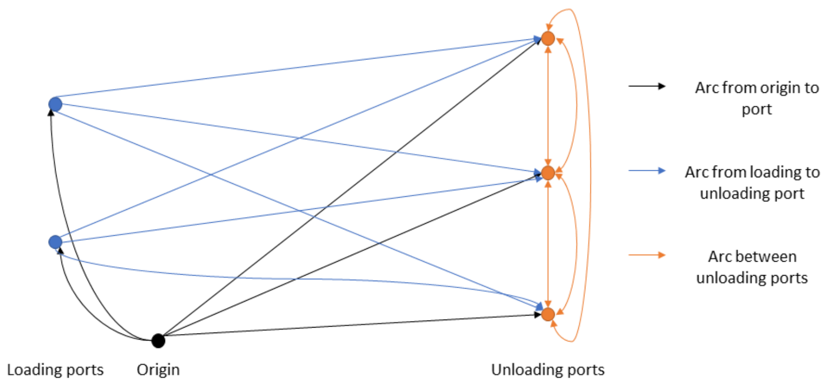

Figure 5.2: Illustration of original network structure

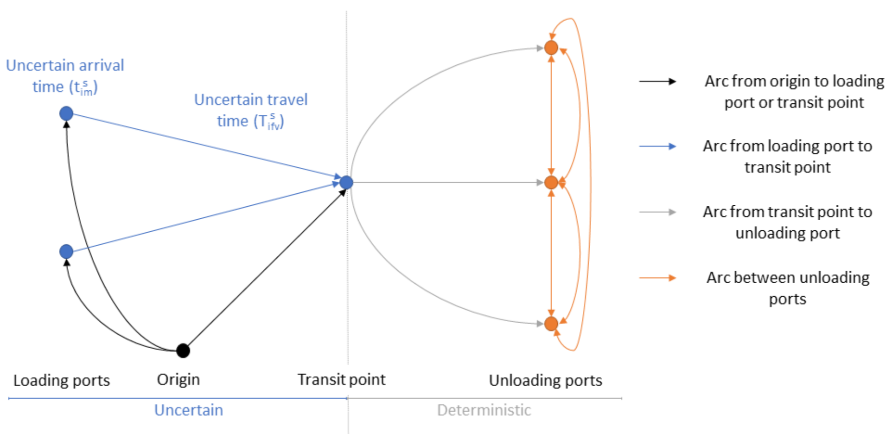

Figure 5.3: Illustration of modified network structure

In the new network structure, the sailing times from the transit point to the unloading ports are short compared to the total travel times. Therefore, we assume that all the uncertainty in travel times is related to the sailing leg from the unloading ports to the transit point. Given this assumption, all uncertainty related to a given ship is revealed before the ship reaches the transit point. Further, we assume that ships travelling directly from origin to the transit point has no associated uncertainty, since they are in transit when the time horizon begins. This fact allows us to re-route the ships after the transit point in stage 2 without considering any uncertainty.

5.3.2 Stage 1 Decisions

As discussed in 5.1, the decisions regarding fleet mix and loading are difficult to change on short notice. However, Hydro has the flexibility to change the routing and unloading decisions. We incorporate this behaviour into our solution framework by fixing the long-term decisions after stage 1 while allowing the routing and unloading decisions to change in stage 2. Specifically, the choice of which ships to use, when to start loading and which loading ports to travel from are fixed in stage 1. On the other hand, the routing and unloading decisions between the transit point and the unloading port are part of the recourse decisions in stage 2.

The approach can be interpreted in the following way. First, we decide that a given ship should pick up its cargo at a given time and sail to the North Sea with a planned sequence of visits to unloading ports. These are the stage 1 decisions that must be determined before the uncertainty is revealed. After a ship arrives at the transit point, Hydro can change its routing to the unloading ports and the quantity unloaded. This interpretation forms the basis for the recourse decisions in stage 2 of our simulation framework.

5.3.3 Mathematical Model Stage 1

Since a continuous time formulation uses less time than a discrete time formulation to solve the problem Nikolaisen and Vågen (2018), the mathematical models for both stages use a continuous time formulation. The mathematical model for stage 1 is the continuous time formulation stated in Nikolaisen and Vågen (2018) with some refinements. Specifically, the valid inequalities in subsection 5.3.4 and constraints (5.17) are added compared to Nikolaisen and Vågen (2018). To simplify, some common notation between the two stages is presented beneath.

Indices

i, j	Port
k	Ship class
m, n	Visit number in port
q	Product quality
v	Ship

Sets

$\mathcal{K}$	Set of ship classes
$\mathcal{M}_i$	Set of possible visits in port i
$\mathcal{P}$	Set of ports
$\mathcal{P}^D$	Set of demand ports, $\mathcal{P}^D \subset \mathcal{P}$
$\mathcal{P}^S$	Set of supply ports, $\mathcal{P}^S \subset \mathcal{P}$
$\mathcal{Q}$	Set of product qualities
$\mathcal{S}^A$	Set of possible nodes (i, m), where (i, m) defines the m^{th} visit to port $i \in \mathcal{P}$
$\mathcal{S}_v^A$	Set of possible nodes (i, m) ship v can visit, $\mathcal{S}_v^A \subset \mathcal{S}^A$
$\mathcal{S}_v^X$	Set of possible sailing arcs (i, m, j, n) for ship v can travel,where (i, m, j, n) defines sailing from node (i, m) to node (j, n) created for $i \in \mathcal{P}$ to $j \in \mathcal{P}^D$ defines sailing from node (i, m) to node (j, n)
$\mathcal{V}$	Set of ships
$\mathcal{V}^k$	Set of ships in class $k \in \mathcal{K}$, $\mathcal{V}^k \subset \mathcal{V}$

Parameters

A_{iq}	Is 1 if port $i \in \mathcal{P}$ can accept or produce quality $q \in \mathcal{Q}$
c_k^T	Cost per time period for using ship class $k \in \mathcal{K}$
c_{ik}^P	Fixed cost of visiting a port $i \in \mathcal{P}$ by ship class $k \in \mathcal{K}$
K_k	Capacity of ship class $k \in \mathcal{K}$
K_v	Capacity of ship $v \in \mathcal{V}$ (equal the class capacity K_k for all ships in class k, $v \in \mathcal{V}^k \Rightarrow K_v = K_k$)
$\overline{L}_i$	Maximum level of total product collected from port $i \in \mathcal{P}^S$
$\underline{L}_i$	Minimum level of total product collected from port $i \in \mathcal{P}^S$
L_i^T	Maximum amount a ship in port $i \in \mathcal{P}$ can (un)load in one time unit
$\overline{N}$	Upper bound on the number of shipments

$\underline{N}$	Lower bound on the number of shipments
Q^O_{vq}	Quantity at ship $v \in \mathcal{V}$ at the beginning of the planning horizon of product quality $q \in \mathcal{Q}$
R_i	Rate of consumption per day at port $i \in \mathcal{P}^D$
$\overline{S}_i$	Maximum stock level at port $i \in \mathcal{P}^D$
$\underline{S}_i$	Minimum stock level at port $i \in \mathcal{P}^D$
$\underline{S}^T_i$	Minimum stock level at port $i \in \mathcal{P}^D$ at the end of the time horizon
S^O_i	Initial stock level at port $i \in \mathcal{P}^D$
T	Length of the time horizon
T_{ijv}	Time required to travel from port $i \in \mathcal{P}$ to port $j \in \mathcal{P}$ for ship $v \in \mathcal{V}$
T^O_{iv}	Travel time from initial position to port $i \in \mathcal{P}$ for ship $v \in \mathcal{V}$
T^S_{iv}	Setup time required in port $i \in \mathcal{P}$ for ship $v \in \mathcal{V}$
$\overline{U}$	Upper bound on total number of demand ports a ship can visit

Decision Variables

f_{imjnvq}	Amount ship $v \in \mathcal{V}$ transports from node $(i, m) \in \mathcal{S}^A$ to node $(j, n) \in \mathcal{S}^A$ of product quality $q \in \mathcal{Q}$
l_{imvq}	Amount loaded or unloaded from ship $v \in \mathcal{V}$ at port $i \in \mathcal{P}$ of product quality $q \in \mathcal{Q}$
s_{im}	Stock level at start of visit $m \in \mathcal{M}_i$ in port $i \in \mathcal{P}^D$
t_{im}	Start time of visit $m \in \mathcal{M}_i$ in port $i \in \mathcal{P}$
t^O_{imv}	Time spent by ship $v \in \mathcal{V}$ operating during visit $m \in \mathcal{M}_i$ to port $i \in \mathcal{P}$
t^W_{imv}	Time spent by ship $v \in \mathcal{V}$ waiting during visit $m \in \mathcal{M}_i$ to port $i \in \mathcal{P}$
x_{imjnv}	Is 1 if ship $v \in \mathcal{V}$ travels arc $(i, m, j, n) \in \mathcal{S}^X_v$, else 0
x^O_{imv}	Is 1 if ship $v \in \mathcal{V}$ travels from initial position to node $(i, m) \in \mathcal{S}^A$, else 0
y_{im}	Is 1 if a ship is making visit $m \in \mathcal{M}_i$ to port $i \in \mathcal{P}$, else 0
z_{imv}	Is 1 if ship $v \in \mathcal{V}$ ends it route at $(i, m) \in \mathcal{S}^A$, else 0
z_v	Is 1 if ship $v \in \mathcal{V}$ is used, else 0
w_{imv}	Is 1 if ship $v \in \mathcal{V}$ is making visit $m \in \mathcal{M}_i$ to port $i \in \mathcal{P}$, else 0

Stage 1 – Objective Function

$$\min Z_1 = \sum_{k\in\mathcal{K}}\sum_{v\in\mathcal{V}^k}\left(c_k^T \cdot \sum_{(i,m,j,n)\in\mathcal{S}_v^X} T_{ijv}\cdot x_{imjnv} + \sum_{(i,m)\in\mathcal{S}_v^A} T_{iv}^O \cdot x_{imv}^O\right) + \sum_{k\in\mathcal{K}}\sum_{v\in\mathcal{V}^k}\sum_{(i,m)\in\mathcal{S}_v^A}\left(t_{imv}^O + t_{imv}^W + \sum_{(i,m)\in\mathcal{S}_v^A} c_{ik}^P \cdot w_{imv}\right) \tag{5.1}$$

The objective function is given by equation (5.1), which minimizes the sum of total cost. The first summand represents the time charter cost when traveling between the ports and from origin, respectively. The second is the time charter cost which occurs during operation and waiting in a port. Further, the last summand in the objective function is the port fees.

Stage 1 – Routing Constraints

$$\sum_{(i,m)\in\mathcal{S}_v^A} x_{imv}^O = z_v, \qquad v\in\mathcal{V} \tag{5.2}$$

$$\underline{N} \le \sum_{v\in\mathcal{V}} z_v \le \overline{N} \tag{5.3}$$

$$\left\lceil \frac{\sum_{q\in\mathcal{Q}} Q_{vq}^0}{K_v} \right\rceil \le z_v, \qquad v\in\mathcal{V} \tag{5.4}$$

$$\sum_{i\in\mathcal{P}^S}\sum_{m\in\mathcal{M}_i} w_{imv} \le z_v, \qquad v\in\mathcal{V} \tag{5.5}$$

$$\sum_{i\in\mathcal{P}^D}\sum_{m\in\mathcal{M}_i} w_{imv} \le \overline{U} z_v, \qquad v\in\mathcal{V} \tag{5.6}$$

$$z_v \ge z_{v+1}, \qquad k\in\mathcal{K}, v\in\mathcal{V}^k \setminus \{|V^k|\} \tag{5.7}$$

Equations (5.2) make sure each ship that is used departs from its initial position for traveling to another node (i, m). Constraints (5.3) ensure that the total number of shipments are in the contracted interval $[\underline{N}, \overline{N}]$ in the planning horizon. Further, Constraints (5.4) make sure that each ship containing an initial load is used.

Equations (5.5) and (5.6) control that only one supply port and a maximum of $\overline{U}$ demand ports are visited, respectively. Further, we introduce the symmetry breaking constraints (5.7) for ship class k. It makes sure that if ship $v+1$ in class $k \in \mathcal{K}$ is used, then ship v in the same class must also be used.

$$\sum_{(j,n)\in\mathcal{S}_v^A} x_{jnimv} + x_{imv}^O = w_{imv}, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A \tag{5.8}$$

$$\sum_{(j,n)\in\mathcal{S}_v^A} x_{imjnv} + z_{imv} = w_{imv}, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A \tag{5.9}$$

$$\sum_{v\in\mathcal{V}} w_{imv} = y_{im}, \qquad (i,m) \in \mathcal{S}^A \tag{5.10}$$

$$y_{i(m-1)} \geq y_{im}, \qquad (i,m) \in \mathcal{S}^A : m > 1 \tag{5.11}$$

Conservation of flow is handled by equations (5.8) and (5.9). Equations (5.8) make sure that a ship travels to node (i,m) if it visits that node and equations (5.9) ensure that a ship either departs from a node or ends it route in it. Moreover, equations (5.10) make sure that a ship can only visit node (i,m) if the variable y_{im} is one. Due to constraints (5.11) port i cannot be visited the m^{th} time if it is not visited in $m-1$.

$$\sum_{i\in\mathcal{P}^D} \sum_{m\in\mathcal{M}^D} x_{imv}^O = \left\lceil \frac{\sum_{q\in\mathcal{Q}} Q_{vq}^O}{K_v} \right\rceil, \qquad v \in \mathcal{V} \tag{5.12}$$

$$z_{imv} = 0, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A : i \in \mathcal{P}^S \tag{5.13}$$

Equations (5.12) make sure each ship that is initial loaded must travel to a demand port, if no load it must begin its trip in a demand node if it is used. Due to constraints (5.13) a ship cannot end its route in a supply node.

 40

Stage 1 – Loading and Unloading Constraints

$$\sum_{q\in\mathcal{Q}} f_{imjnvq} \leq K_v x_{imjnv}, \qquad v\in\mathcal{V}, (i,m,j,n)\in\mathcal{S}_v^X \tag{5.14}$$

$$\sum_{q\in\mathcal{Q}} l_{imvq} \leq \min\{K_v, \overline{S}_i - \underline{S}_i\} w_{imv}, \quad v\in\mathcal{V}, (i,m)\in\mathcal{S}_v^A : i\in\mathcal{P}^D \tag{5.15}$$

$$\sum_{q\in\mathcal{Q}} l_{imvq} \leq \min\{K_v, \overline{L}_i\} w_{imv}, \qquad v\in\mathcal{V}, (i,m)\in\mathcal{S}_v^A : i\in\mathcal{P}^S \tag{5.16}$$

$$\sum_{q\in\mathcal{Q}} l_{imvq} \geq K_v w_{imv}, \qquad v\in\mathcal{V}, (i,m)\in\mathcal{S}_v^A : i\in\mathcal{P}^S \tag{5.17}$$

$$l_{imvq} \leq A_{iq} K_v w_{imv}, \qquad v\in\mathcal{V}, (i,m)\in\mathcal{S}_v^A, q\in\mathcal{Q} \tag{5.18}$$

Constraints (5.14) ensure that the quantity on board ship v never exceeds maximum capacity. Further, equations (5.15) and (5.16) state that quantity loaded or unloaded on ship v is below the minimum of the ship's capacity and what is allowed in port for the demand ports and supply ports, respectively. Constraints (5.17) ensure that each ship load leaving a supply port is fully loaded. Furthermore, constraints (5.18) make sure that no ship can load or unload a quality that is either not produced in the loading port or not accepted in the unloading port.

$$Q_{vq}^O x_{imv}^O + l_{imvq} = \sum_{(j,n)\in\mathcal{S}_v^A} \left(f_{imjnvq} - f_{jnimvq}\right), \qquad v\in\mathcal{V}, q\in\mathcal{Q}, \; (i,m)\in\mathcal{S}_v^A : i\in\mathcal{P}^S \tag{5.19}$$

$$Q_{vq}^O x_{imv}^O - l_{imvq} = \sum_{(j,n)\in\mathcal{S}_v^A} \left(f_{imjnvq} - f_{jnimvq}\right), \qquad v\in\mathcal{V}, q\in\mathcal{Q}, \; (i,m)\in\mathcal{S}_v^A : i\in\mathcal{P}^D \tag{5.20}$$

Equations (5.19) and (5.20) represent the mass conservation for node (i,m) in loading ports and unloading ports, respectively.

Stage 1 – Time Constraints

$$\frac{\sum_{q\in\mathcal{Q}} l_{imvq}}{L_i^T} + T_{iv}^S w_{imv} \leq t_{imv}^O, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A \tag{5.21}$$

$$t_{imv}^W + t_{imv}^O \leq T w_{imv}, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A \tag{5.22}$$

Constraints (5.21) make sure that the operational time in a port is greater than the setup time in a port in addition to the time needed to (un)load the ship. Further, Constraints (5.22) ensure that the operational and waiting time for ship v in port i are zero, if ship v is not operating.

$$t_{i,m-1} + \sum_{v\in\mathcal{V}} t_{i,m-1,v}^o \leq t_{im}, \qquad (i,m) \in \mathcal{S}^A : m > 1 \tag{5.23}$$

$$\sum_{v\in\mathcal{V}} \left(T_{iv}^O x_{imv}^O + t_{imv}^W\right) \leq t_{im}, \qquad (i,m) \in \mathcal{S}^A \tag{5.24}$$

$$(T - T_{iv}^O) x_{imv}^O + t_{imv}^W \leq T - t_{im} \qquad \begin{array}{l}(i,m) \in \mathcal{S}^A : i \in \mathcal{P}^D, \\ v \in \mathcal{V} : \sum_{q\in\mathcal{Q}} Q_{vq}^O > 0\end{array} \tag{5.25}$$

$$t_{im} + \sum_{v\in\mathcal{V}} t_{imv}^O \leq T, \qquad (i,m) \in \mathcal{S}^A \tag{5.26}$$

Due to constraints (5.23), only one ship is allowed to be at port i at the same time. The constraints (5.24) assign the start time in the first node to be at least the travel time from the initial position for ship v to the first port. Constraints (5.25) make sure that ships that are in transit at the beginning of the horizon leave the origin at time zero. Constraints (5.26) ensures that the start time of visits and operational time for the visit is within the planning horizon.

$$t_{im} + \sum_{v\in\mathcal{V}} \left(t_{imv}^O + t_{jnv}^W + (T_{ijv} + T) \cdot x_{imjnv}\right) \leq t_{jn} + T, \quad (i,m),(j,n) \in \mathcal{S}^A \tag{5.27}$$

$$t_{im} + \sum_{v\in\mathcal{V}} \left(t_{imv}^O + t_{jnv}^W + (T_{ijv} - T) \cdot x_{imjnv}\right) \geq t_{jn} - T, \quad (i,m),(j,n) \in \mathcal{S}^A \tag{5.28}$$

 42

The last time constraints given by (5.27)-(5.28) connect the start time at node (i, m) with the start time in (j, n) when ship v directly travels to (j, n) from (i, m).

Stage 1 – Inventory Constraints

$$s_i^O - s_{i1} = R_i t_{i1}, \qquad i \in \mathcal{P}^D \tag{5.29}$$

$$s_{i(m-1)} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}} l_{i,m-1,vq} - s_{im} = R_i \big(t_{im} - t_{i,m-1}\big), \qquad i \in \mathcal{P}^D, \; m \in \mathcal{M}_i \setminus \{1\} \tag{5.30}$$

Equations (5.29) set the initial stock levels. Further, the stock levels in the beginning of the m^{th} visit are related to the visit before in equations (5.30).

$$s_{im} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}} l_{imvq} - \sum_{v \in \mathcal{V}} R_i t_{imv}^O \leq \overline{S}_i, \qquad i \in \mathcal{P}^D, m \in \mathcal{M}_i \tag{5.31}$$

$$s_{im} \geq \underline{S}_i, \qquad i \in \mathcal{P}^D, m \in \mathcal{M}_i \tag{5.32}$$

$$s_{i\overline{M}_i} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}} l_{i,\overline{M}_i,vq} - R_i\big(T - t_{i\overline{M}_i}\big) \geq \underline{S}_i^T, \qquad i \in \mathcal{P}^D \tag{5.33}$$

$$\underline{L}_i \leq \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}} \sum_{m \in \mathcal{M}_i} l_{imvq} \leq \overline{L}_i, \qquad i \in \mathcal{P}^S \tag{5.34}$$

Constraints (5.31) make sure that the maximum stock level in demand port i is not exceeded. On the other hand, constraints (5.32) and (5.33) impose a lower inventory bound in demand port i for the inventory level at the beginning of visit m and at the end of the time horizon, respectively. Lastly, the total volume supplied from supply port i is ensured to be within the given interval $[\underline{L}_i, \overline{L}_i]$ by constraints (5.34).

Stage 1 – Non-negativity and Binary Restrictions

$$f_{imjnvq} \geq 0, \qquad v \in \mathcal{V}, (i,m,j,n) \in \mathcal{S}_v^X, q \in \mathcal{Q} \tag{5.35}$$

$$l_{imvq} \geq 0, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A, q \in \mathcal{Q} \tag{5.36}$$

$$t_{im} \geq 0, \qquad (i,m) \in \mathcal{S}^A \tag{5.37}$$

$$t_{imv}^W, t_{imv}^O \geq 0, \qquad (i,m) \in \mathcal{S}^A, v \in \mathcal{V} \tag{5.38}$$

$$x_{imjnv} \in \{0,1\}, \qquad v \in \mathcal{V}, (i,m,j,n) \in \mathcal{S}_v^X \tag{5.39}$$

$$x_{imv}^O, w_{imv}, z_{imv} \in \{0,1\}, \qquad v \in \mathcal{V}, (i,m) \in \mathcal{S}_v^A \tag{5.40}$$

$$y_{im} \in \{0,1\}, \qquad (i,m) \in \mathcal{S}^A \tag{5.41}$$

$$z_v \in \{0,1\}, \qquad v \in \mathcal{V} \tag{5.42}$$

Constraints (5.35) - (5.38) make sure that the loading, quantity and time variables are non-negative. Lastly, all binary routing variables are defined as binary in constraints (5.39) - (5.42).

5.3.4 Formulation of Valid Inequalities

In the MIRP literature, several ways of further strengthening and tightening the arc-load-flow models have been identified. We will in this subsection introduce the valid inequalities we have implemented.

The first two presented impose a minimum number of visits to each port and a minimum total operational time in each port, respectively. They are similar to Agra et al. (2016) which is also presented in Nikolaisen and Vågen (2018). The third presented impose an upper limit on start time of operation for each visit, t_{im}, inspired by Agra et al. (2013b).

Like Nikolaisen and Vågen (2018) we will simplify the notation by introducing the parameters: unloading port's net demand over the total time horizon, minimum number of visits and minimum operational time. The latter two are introduced in each port. Net demand over the total time horizon for each unloading port can be calculated as $ND_i = \max\{T \cdot R_i - S_i^0 + \underline{S}_i, 0\}, i \in \mathcal{P}^D$. The minimum number of visits in port $i \in \mathcal{P}^D$ is given by $\underline{M}_i = \left\lceil \frac{ND_i}{\max\{K_k\}} \right\rceil$ and the minimum operational time in port $i \in \mathcal{P}^D$ is given by $\underline{T}_i^O = \frac{ND_i}{L_i^T} + \underline{M}_i \cdot T_i^F$.

For the loading ports, $\underline{M}_i = \lceil \frac{\underline{L}_i}{\max\{K_k\}} \rceil$ and $\underline{T}_i^O = \frac{\underline{L}_i}{L_i^T} + \underline{M}_i \cdot T_i^F$ are defined over the entire time horizon using the lower bound for quantity supplied, $\underline{L}_i$ for $i \in \mathcal{P}^S$.

Minimum Visits and Minimum Operational Time in each Port

Agra et al. (2016) present valid inequalities for minimum number of visits and minimum operational time in port. For the model, minimum visits can only be imposed over the total time horizon. Therefore, we implement the two constraints beneath, where (5.43) represent the minimum visits in port $i \in \mathcal{P}$ and (5.44) represent minimum operational time in port $i \in \mathcal{P}$.

$$y_{im} = 1, \qquad i \in \mathcal{P}, m \leq \underline{M}_i \tag{5.43}$$

$$\sum_{m \in \mathcal{M}_i} \sum_{v \in \mathcal{V}} t_{imv}^O \geq \underline{T}_i^O, \qquad i \in \mathcal{P} \tag{5.44}$$

Upper Limit on Time Visited Variables for each Visit

The last valid inequalities is inspired by Agra et al. (2013b). It imposes a maximum limit on the time visited variables in the unloading ports by utilizing the storage capacity as well as the consumption rate. In port $i \in \mathcal{P}^D$ let the last possible time for visit $m \in \mathcal{M}_i$ be T_{im}^{MAX}, which is defined as follows:

$$T_{im}^{MAX} = \min \left\{ T \, , \frac{S_i^0 + (m-1) \cdot \overline{S}_i - \underline{S}_i}{R_i} - T_i^F \right\}, \qquad (i,m) \in \mathcal{S}^A : i \in \mathcal{P}^D$$

This gives the following inequality constraints:

$$t_{im} \leq T_{im}^{MAX}, \qquad (i,m) \in \mathcal{S}^A : i \in \mathcal{P}^D \tag{5.45}$$

5.4 Stage 2 – Evaluation of Potential Solutions

The objective in stage 2 is to evaluate the planned schedule, also referred to as the potential solution, generated in stage 1. The performance of each potential solution is evaluated over $s \in \mathcal{S}$ simulations. In each simulation the uncertain travel times and delays in loading ports are drawn randomly for each ship. Then, the stage 2 re-routing problem is solved to get an estimate of the total cost of this particular realization of the uncertain parameters. Finally, the cost of uncertainty associated with the stage 1 solution is estimated as the difference between the average realized costs in stage 2 over all simulations and the planned cost from stage 1. This sections begins by describing the stage 2 optimization problem in subsection 5.4.1 before the mathematical model is presented in subsection 5.4.2.

5.4.1 Stage 2 Problem Description

The stage 2 re-routing problem is similar to the stage 1 problem in the sense that it is still a MIRP with the same network of unloading ports. A set of ships start at the transit points and are routed to deliver their cargo to the unloading ports in such a way that the inventory bounds are not violated. The network structure for this problem is illustrated in Figure 5.4.

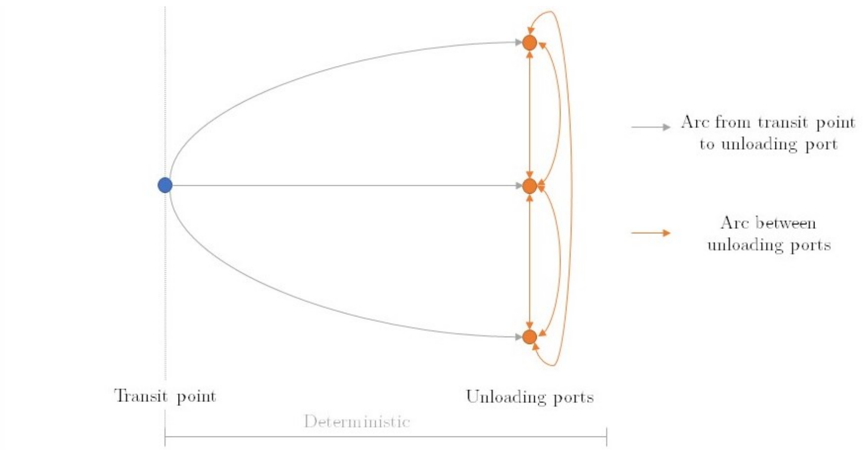

Figure 5.4: Illustration of stage 2 network structure

There are several notable differences compared to the stage 1 problem. Firstly, only unloading is considered. The ships start fully loaded in the transit point and are not allowed to pick up any cargo at the ports. The re-routing model decides which unloading ports a ship will visit, its visit order as well as the quantity unloaded. Secondly, the time at which a ship must leave the transit point is fixed based on the planned arrival time from stage 1 and the realized delays in the simulation. Thirdly, all the ships that reach the transit point are already chartered. Therefore, the full set of chartered ships in the stage 1 solution needs to be used in the stage 2 re-routing model.

 46

5.4.2 Mathematical Model for the Stage 2 Re-Routing Problem

For constructing the mathematical model for stage 2 some of the notation is re-defined as follow.

Additional Sets

$\mathcal{S}^{A2}$	Set of possible nodes (i,m), where (i,m) defines the m^{th} visit to port $i \in \mathcal{P}^D$
$\mathcal{S}_v^{A2}$	Set of possible nodes (i,m) ship v can visit, $\mathcal{S}_v^{A2} \subset \mathcal{S}_v^A$
$\mathcal{S}_v^{X2}$	Set of possible sailing arcs (i,m,j,n) for ship v can travel,where (i,m,j,n) defines sailing from node (i,m) to node (j,n), $\mathcal{S}_v^{X2} \subset \mathcal{S}_v^X$
$\mathcal{V}^2$	Set of ships used in stage 1, $\mathcal{V}^2 \subset \mathcal{V}$

Additional Parameters

T_{fv}	The time ship $v \in \mathcal{V}^2$ leaves the transit point

Additional Variables

t_{fv}	The time ship $v \in \mathcal{V}^2$ leaves the transit point
x_{imv}^f	Is 1 if ship $v \in \mathcal{V}^2$ visits node $(i,m) \in \mathcal{S}_v^{A2}$, else 0

Stage 2 − Objective Function

$$\begin{aligned}\min Z_2 = &\sum_{v\in\mathcal{V}^2}\left(c_v^T \cdot \sum_{(i,m,j,n)\in\mathcal{S}_v^{X2}} T_{ijv}\cdot x_{imjnv} + \sum_{(i,m)\in\mathcal{S}_v^{A2}} T_{iv}^O \cdot x_{imv}^O\right)\\ &+ \sum_{v\in\mathcal{V}^2}\sum_{(i,m)\in\mathcal{S}_v^{A2}}\left(t_{imv}^O + t_{imv}^W + \sum_{(i,m)\in\mathcal{S}_v^{A2}} c_{iv}^P \cdot w_{imv}\right)\end{aligned} \tag{5.46}$$

The objective is given by equation (5.46), and is similar to the one presented in the original model for stage 1.

Stage 2 – Routing Constraints

$$\sum_{(i,m)\in\mathcal{S}_v^{A2}} x_{imv}^f = 1, \qquad v \in \mathcal{V}^2 \tag{5.47}$$

$$\sum_{(i,m)\in\mathcal{S}^{A2}} w_{imv} \leq \overline{U} z_v, \qquad v \in \mathcal{V}^2 \tag{5.48}$$

$$\sum_{(j,n)\in\mathcal{S}_v^{A2}} x_{jnimv} + x_{imv}^f = w_{imv}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.49}$$

$$\sum_{(j,n)\in\mathcal{S}_v^{A2}} x_{imjnv} + z_{imv} = w_{imv}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.50}$$

$$\sum_{v\in\mathcal{V}^2} w_{imv} = y_{im}, \qquad (i,m) \in \mathcal{S}_v^{A2} \tag{5.51}$$

$$y_{i(m-1)} \geq y_{im}, \qquad (i,m) \in \mathcal{S}^{A2} : m > 1 \tag{5.52}$$

Equations (5.47) make sure each ship departs from its initial position, and represent a modification of constraint (5.2), while the rest of the given constraints are equal to constraints (5.2), (5.6) and (5.8) - (5.11) in stage 1 model for $i \in \mathcal{P}^D$, respectively.

Stage 2 – Loading and Unloading Constraints

$$\sum_{q\in\mathcal{Q}} l_{imvq} \leq \min\{K_v, \overline{S}_i - \underline{S}_i\} w_{imv}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.53}$$

$$l_{imvq} \leq A_{iq} K_v w_{imv}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2}, q \in \mathcal{Q} \tag{5.54}$$

$$Q_{vq}^O x_{imv}^f - l_{imvq} = \sum_{(j,n)\in\mathcal{S}_v^{A2}} \left(f_{imjnvq} - f_{jnimvq}\right), \quad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2}, q \in \mathcal{Q} \tag{5.55}$$

The given constraints are equal to constraints (5.15), (5.18) and (5.20) in stage 1 for $i \in \mathcal{P}^D$, respectively.

48

Stage 2 – Time Constraints

$$\frac{\sum_{q\in\mathcal{Q}} l_{imvq}}{L_i^T} + T_{iv}^S w_{imv} \leq t_{imv}^O, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.56}$$

$$t_{imv}^W + t_{imv}^O \leq T w_{imv}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.57}$$

$$t_{i,m-1} + \sum_{v\in\mathcal{V}^2} t_{i,m-1,v}^o \leq t_{im}, \qquad (i,m) \in \mathcal{S}^{A2} : m > 1 \tag{5.58}$$

$$t_{im} + \sum_{v\in\mathcal{V}^2} t_{imv}^O \leq T, \qquad (i,m) \in \mathcal{S}^{A2} \tag{5.59}$$

$$t_{im} + \sum_{v\in\mathcal{V}^2} \left(t_{imv}^O + t_{jnv}^W + (T_{ijv} + T)\cdot x_{imjnv}\right) \leq t_{jn} + T, \quad (i,m),(j,n) \in \mathcal{S}^{A2} \tag{5.60}$$

$$t_{im} + \sum_{v\in\mathcal{V}^2} \left(t_{imv}^O + t_{jnv}^W + (T_{ijv} - T)\cdot x_{imjnv}\right) \geq t_{jn} - T, \quad (i,m),(j,n) \in \mathcal{S}^{A2} \tag{5.61}$$

The constraints above are the same as constraints (5.21) - (5.23) and (5.26) - (5.28) in stage 1 for $i \in \mathcal{P}^D$, respectively.

$$t_{fv} = T_{fv}, \qquad v \in \mathcal{V}^2 \tag{5.62}$$

$$t_{fv} + t_{imv}^W + \left(T_{iv}^f + T\right) \cdot x_{imv}^f \leq t_{im} + T, \quad (i,m),(j,n) \in \mathcal{S}^{A2}, v \in \mathcal{V}^2 \tag{5.63}$$

$$t_{fv} + t_{imv}^W + \left(T_{iv}^f - T\right) \cdot x_{imv}^f \geq t_{im} - T, \quad (i,m),(j,n) \in \mathcal{S}^{A2}, v \in \mathcal{V}^2 \tag{5.64}$$

New time constraints compared to stage 1 are given above. The first, constraints (5.62) say that the leaving time from the transit point must be equal to the one given in stage 1 after uncertainty in travel time and arrival time at loading port is revealed. Therefore, these constraints separates the different scenarios according to the uncertainty generated. Hence, it is the reason for the different stage 2 solutions. Further, constraints (5.63)-(5.64) connect the start time at node (i, m) with the leaving time from the transit point when the ship travels directly to (i, m) from transit.

Stage 2 – Inventory Constraints

$$s_i^O - s_{i1} = R_i t_{i1}, \qquad i \in \mathcal{P}^D \tag{5.65}$$

$$s_{i(m-1)} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}^2} l_{i,m-1,vq} - s_{im} = R_i\big(t_{im} - t_{i,m-1}\big), \qquad i \in \mathcal{P}^D, \; m \in \mathcal{M}_i \setminus \{1\} \tag{5.66}$$

$$s_{im} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}^2} l_{imvq} - \sum_{v \in \mathcal{V}^2} R_i t_{imv}^O \le \overline{S}_i, \qquad i \in \mathcal{P}^D, m \in \mathcal{M}_i \tag{5.67}$$

$$s_{im} \ge \underline{S}_i, \qquad i \in \mathcal{P}^D, m \in \mathcal{M}_i \tag{5.68}$$

$$s_{i\overline{M}_i} + \sum_{q \in \mathcal{Q}} \sum_{v \in \mathcal{V}^2} l_{i,\overline{M}_i,vq} - R_i\big(T - t_{i\overline{M}_i}\big) \ge \underline{S}_i^T, \qquad i \in \mathcal{P}^D \tag{5.69}$$

Constraints (5.65) - (5.69) are the same as constraints (5.29) - (5.33) in stage 1, respectively.

Stage 2 – Non-negativity and Binary Restrictions

$$f_{imjnvq} \ge 0, \qquad v \in \mathcal{V}^2, (i,m,j,n) \in \mathcal{S}_v^{X2}, q \in \mathcal{Q} \tag{5.70}$$

$$l_{imvq} \ge 0, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2}, q \in \mathcal{Q} \tag{5.71}$$

$$t_{fv} \ge 0, \qquad v \in \mathcal{V}^2 \tag{5.72}$$

$$t_{im} \ge 0, \qquad (i,m) \in \mathcal{S}^{A2} \tag{5.73}$$

$$t_{imv}^W, t_{imv}^O \ge 0, \qquad (i,m) \in \mathcal{S}^{A2}, v \in \mathcal{V}^2 \tag{5.74}$$

$$x_{imjnv} \in \{0,1\}, \qquad v \in \mathcal{V}^2, (i,m,j,n) \in \mathcal{S}_v^{X2} \tag{5.75}$$

$$x_{imv}^f, w_{imv}, z_{imv} \in \{0,1\}, \qquad v \in \mathcal{V}^2, (i,m) \in \mathcal{S}_v^{A2} \tag{5.76}$$

$$y_{im} \in \{0,1\}, \qquad (i,m) \in \mathcal{S}^{A2} \tag{5.77}$$

Constraints (5.70) - (5.74) make sure that the quantities transported, quantities unloaded and time variables are non-negative. Lastly, all binary routing variables are defined as binary in constraints (5.75) - (5.77).

Stage 2 – Valid Inequalities

The following valid inequalities is also introduced for stage 2:

$$y_{im} = 1, \qquad i \in \mathcal{P}^D, m \leq \underline{M}_i \tag{5.78}$$

$$\sum_{m \in \mathcal{M}_i} \sum_{v \in \mathcal{V}^2} t_{imv}^o \geq \underline{T}_i^O, \qquad i \in \mathcal{P}^D \tag{5.79}$$

$$t_{im} \leq T_{im}^{MAX}, \qquad (i,m) \in \mathcal{S}^A \tag{5.80}$$

The valid inequalities presented by constraints (5.78)-(5.80) are the same as constraints (5.43)-(5.45), in stage 1, respectively.

5.5 Stage 3 – Feedback Process

As mentioned in section 5.2, our solution framework iterates between creating a potential solution in stage 1 and evaluating it under uncertainty in stage 2. After stage 2, an estimated cost of uncertainty is associated with the potential solution and the process is repeated until stage 1 generates a solution that has been evaluated before.

This section begins by discussing how the framework can approximate an optimal solution under uncertainty in subsection 5.5.1. Thereafter, we explain how to associate a penalty cost with previously tested stage 1 solutions in subsections 5.5.2 and 5.5.3. Further, the mathematical implementation of the feedback loop is presented in subsection 5.5.4. Finally, we discuss how to handle sub-optimal solutions from stage 1 in subsection 5.5.5.

5.5.1 Achieving Optimality under Uncertainty

The feedback process converges towards a solution that is expected to be optimal under uncertainty if the cost of uncertainty is positive. Under this assumption, uncertainty always increases the cost of a potential solution. Therefore, the deterministic optimal solution forms a lower bound on the costs under uncertainty. Furthermore, the realized cost of all previous solutions evaluated form upper bounds on the objective value in stage 1. Once a solution that has previously been evaluated is re-chosen including its cost of uncertainty, we can conclude that this is the optimal solution under uncertainty. This solution has a lower realized cost than the planned cost of any other potential solution.

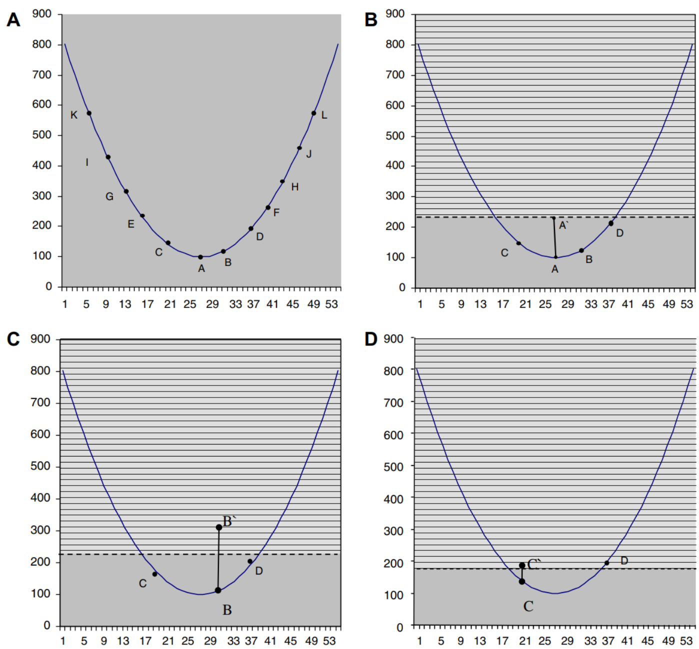

Figure 5.5: Illustration of iterative solution method. In graphs A-D, the x-axis represents variable values and the y-axis represents objective values (Acar et al. 2009)

Figure 5.5 illustrates this process. In the first iteration, solution A is optimal. After the cost of uncertainty is added to solution A, the deterministic cost of solution B is the new optimal stage 1 solution. In iteration 3, solution C is chosen. The algorithm will terminate in the next iteration because the true cost of solution C is lower than the deterministic cost of solution D and lower than the true cost of A and B.

If the cost of uncertainty can be negative, we cannot guarantee optimality under uncertainty. The realized costs of all previous solutions still form upper bounds on the the objective value in stage 1. However, the best deterministic solution no longer forms a lower bound. The framework terminates if it finds a solution with realized costs lower than the deterministic optimal costs even though there may exist better solutions that have not been explored. This represents a problem if the uncertainty often leads to improved solutions.

In this thesis, there is a possibility of getting negative costs of uncertainty. Specifically, it can happen if a ship that begins loading very early has a shorter travel time than expected. This event expands the solution space of the stage 2 model compared to the stage 1 model and may therefore lead to lower costs. However, the probability of this event is low and the possible cost reductions are small compared to the cost increases of stock-outs and delays. Therefore, we consider the risk of missing the best solutions acceptable.

5.5.2 Method for Associating a Stage 1 Solution with its Penalty Cost

This subsection restates the method introduced by Acar et al. (2009) for associating the stage 1 solutions with their respectively penalty cost. They consider a general MIP with binary variables x_i. Further, let $Y_{\eta i}$ be the value of binary variable x_i in candidate solution η and let M be a large number. The following constraints ensures that $\lambda_\eta = 1$ if the current solution is equal to candidate solution η (Acar et al. 2009).

$$\sum_i (2Y_{\eta i} \cdot x_i - Y_{\eta i} - x_i) \leq \lambda_\eta - 1 \tag{5.81}$$

$$\sum_i (2Y_{\eta i} \cdot x_i - Y_{\eta i} - x_i) \geq M(\lambda_\eta - 1) \tag{5.82}$$

5.5.3 Variables Defining a Stage 1 Solution

The stage 1 solution is defined by the variables that are not allowed to change in stage 2, namely the fleet mix and the decisions in the unloading ports. The variables given beneath together determine the timing of loading, the quantity of loading, which ships that are chartered and the planned arrival time for all ships in the transit point. These variables define a stage 1 solution and we believe they use as few variables as possible to do so. It is desirable to use few variables in order to limit the number of different candidate solutions that can be created.

- $\sum_{v \in \mathcal{V}^k} w_{imv}$ for $i \in \mathcal{P}^S, m \in \mathcal{M}_i, k \in \mathcal{K}$ - The class of ship making a specific visit in an loading port
- z_v for $v \in \mathcal{V}$ - The ships used in a solution
- t_{im} for $i \in \mathcal{P}^S, m \in \mathcal{M}_i$ - The time a ship plans to visit a loading port

The method described in subsection 5.5.2 only includes binary variables, but in our formulation the timing of visits, t_{im}, are continuous variables. If we implement the method by Acar et al. (2009) directly, it will not converge because there exists an infinite number of different solutions where the continuous variables are marginally altered. Therefore, we choose to discretize the time horizon into intervals $\tau \in \mathcal{T}$ of length ΔT. The binary variable $\gamma_{im\tau}$ is 1 if node (i, m) is visited in time interval τ. This discretization is used to determine whether or not two solutions are identical and is necessary for the framework to converge.

At this point, it is important to note that the solution method is sensitive to the length of the time intervals ΔT. Choosing large ΔT helps the method converge quicker. However, parts of the solution space is excluded when $\Delta T > 0$. Solutions that potentially are optimal may not be evaluated because they are deemed to be identical to a previously tested solution. Thus, we cannot guaranty that the optimal solution is found when the framework converges. The choice of ΔT represents a trade-off. If ΔT is too large, certain areas of the solution space will go unexplored and there is a risk of missing good solutions. However, if ΔT is too small, the framework will not converge because too many similar solutions are evaluated.

Furthermore, note that this discretization of the time windows for visits in loading ports is not the same as using a discrete time formulation as described by Nikolaisen and Vågen (2018) since time is not indexing each variable in the mathematical formulation.

5.5.4 A Mathematical Implementation of the Feedback Loop

The feedback loop is introduced to estimate how the stage 1 solutions are able to handle uncertainty. To be able to implement a feedback loop some additional modifications must be done. Hence, we need to introduce some additional notation.

Additional Indices

η	Stage 1 solution number in feedback loop
τ	Time interval number

Additional Sets

$\mathcal{H}$	Set of solutions found in earlier feedback loops
$\mathcal{T}$	Set of time intervals

Additional Parameters

ΔC_η	Estimated difference in cost between simulation of solution and deterministic solution from stage 1 for potential solution $\eta \in \mathcal{H}$
$M^A_{i\eta}$	Actual number of visits done in loading port $i \in \mathcal{P}^S$ in stage 1 solution number $\eta \in \mathcal{H}$
$\Gamma_{im\tau\eta}$	Is 1 if node $(i, m) \in \mathcal{S}^A : i \in \mathcal{P}^S$ is served in time interval τ in stage 1 solution number $\eta \in \mathcal{H}$
$W_{imk\eta}$	Is 1 if ship class $k \in \mathcal{K}$ is used to serve node $(i, m) \in \mathcal{S}^A : i \in \mathcal{P}^S$ in stage 1 solution number $\eta \in \mathcal{H}$
$Z_{v\eta}$	Is 1 if ship v is used in stage 1 solution number $\eta \in \mathcal{H}$
ΔT	Is the length of each time interval, $\Delta T = \frac{T}{\lvert\mathcal{T}\rvert}$

Decision Variables

λ_η	Is 1 if same solution is found as stage 1 solution number $\eta \in \mathcal{H}$
$\gamma_{im\tau}$	Is 1 if node $(i, m) \in \mathcal{S}^A : i \in \mathcal{P}^S$ is visited in time interval $\tau \in \mathcal{T}$

Additional Constraints

$$t_{im} \geq (\tau - 1) \cdot \Delta T \cdot \gamma_{im\tau}, \qquad i \in \mathcal{P}^S, m \in \mathcal{M}_i, \tau \in \mathcal{T} \tag{5.83}$$

$$t_{im} - \tau \Delta T \leq (T - \tau \Delta T)(1 - \gamma_{im\tau}), \quad i \in \mathcal{P}^S, m \in \mathcal{M}_i, \tau \in \mathcal{T} \tag{5.84}$$

$$\sum_{\tau \in \mathcal{T}} \gamma_{im\tau} = 1, \qquad i \in \mathcal{P}^S, \eta \in \mathcal{H}, m \in 1, ..., M^A_{i\eta} \tag{5.85}$$

In order to discretize the time windows for visits in loading ports constraints (5.83)-(5.85) are implemented. They ensure that the binary variable used to identify the time window is set to 1 if it is visited in the given window, else zero.

$$\begin{aligned}&\sum_{i\in\mathcal{P}^S}\sum_{m=1}^{M_{i\eta}^A+1}\Bigg(\sum_{k\in\mathcal{K}}2W_{imk\eta}\cdot\sum_{v\in\mathcal{V}^k}w_{imv}-W_{imk\eta}\\&\qquad-\sum_{v\in\mathcal{V}^k}w_{imv}\Bigg)+\sum_{v\in\mathcal{V}}\Big(2Z_{v\eta}\cdot z_v-Z_{v\eta}-z_v\Big)\\&+\sum_{i\in\mathcal{P}^S}\sum_{m=1}^{M_{i\eta}^A+1}\sum_{\tau\in\mathcal{T}}\Big(2\Gamma_{im\tau\eta}\cdot\gamma_{im\tau}-\Gamma_{im\tau\eta}-\gamma_{im\tau}\Big)\\&\geq M\cdot(\lambda_\eta-1)\end{aligned}\quad,\eta\in\mathcal{H} \tag{5.86}$$

$$\begin{aligned}&\sum_{i\in\mathcal{P}^S}\sum_{m=1}^{M_{i\eta}^A+1}\Bigg(\sum_{k\in\mathcal{K}}2W_{imk\eta}\cdot\sum_{v\in\mathcal{V}^k}w_{imv}-W_{imk\eta}\\&\qquad-\sum_{v\in\mathcal{V}^k}w_{imv}\Bigg)+\sum_{v\in\mathcal{V}}\Big(2Z_{v\eta}\cdot z_v-Z_{v\eta}-z_v\Big)\\&+\sum_{i\in\mathcal{P}^S}\sum_{m=1}^{M_{i\eta}^A+1}\sum_{\tau\in\mathcal{T}}\Big(2\Gamma_{im\tau\eta}\cdot\gamma_{im\tau}-\Gamma_{im\tau\eta}-\gamma_{im\tau}\Big)\\&\leq \lambda_\eta-1\end{aligned}\quad,\eta\in\mathcal{H} \tag{5.87}$$

Further, we need to check if a solution has been tested before. Constraints (5.86)-(5.87) ensure the variable $\lambda_\eta = 1$ if the solution is found in an earlier loop. We then define the big M-parameter as follows:

$$M = 2\cdot\min\{\sum_{i\in\mathcal{P}^S} M_i, \overline{N}\} + \overline{N} \tag{5.88}$$

In addition to the objective already defined in equation (5.1) a penalty cost for choosing an optimal solution from a previous loop is needed. Therefore, the new objective function is defined by equation (5.89).

$$\min Z_1^{'} = Z_1 + \sum_{\eta\in\mathcal{H}} \Delta c_\eta \cdot \lambda_\eta \tag{5.89}$$

The last summand in the objective function (5.89) represent a penalty cost for choosing a solution already used in a previous loop.

Due to the results from the previous loops an upper bound on the objective function are also fed into the stage 1 optimization model. Therefore, the following constraint also holds:

$$Z_1^{'} \leq \min{}_{\eta \in \mathcal{H}} \big(Z_{1,\eta}^{'*} + \Delta C_\eta \big) \tag{5.90}$$

Please note that while the set $\mathcal{T}$ is constant throughout the feedbackloop, the set of loops executed in the feedbackloop, $\mathcal{H}$, is empty ($\mathcal{H} = \emptyset$) in the initial run of stage 1. Thereafter, it increases in size for each iteration with the number of solutions extracted from the previous stage 1 run.

5.5.5 Ensuring Correct Feedback from Evaluation of Suboptimal Stage 1 Solutions

For sub-optimal solutions from stage 1, there is a risk of miscalculating the cost of uncertainty in stage 2. The cost associated with the solution found in stage 1, Z_1, may be sub-optimal because of costly decisions before or after the transit point. If the decisions after the transit point are poor, they may be improved in stage 2, leading to lower costs. Therefore, the resulting cost of uncertainty will be very low, possibly even negative. In the next iteration of stage 1, the model can associate the very low penalty cost with a solution with the same stage 1 decisions but better decisions after the transit point. The net effect is that the wrong penalty cost is associated with the latter stage 1 solution, giving it an artificially low total cost.

Therefore, each potential solution extracted from stage 1 is run initially with a deterministic scenario without any delays in stage 2. The result from the deterministic simulation is the optimal solution, Z_1^*, given the stage 1 decisions. The correct cost of uncertainty is calculated as (5.91). Here, $Z_{1\eta}^*$ is the stage 1 optimal deterministic cost for the solution $\eta \in \mathcal{H}$, $\mathrm{E}[Z_{2\eta}^*]$ is the estimated cost for the stage 2 solution, while ΔC_η is the estimated cost of uncertainty for the potential solution.

$$\Delta C_\eta = \mathrm{E}[Z_{2\eta}^*] - Z_{1\eta}^*, \qquad \eta \in \mathcal{H} \tag{5.91}$$

Chapter 6

Extensions to the Solution Framework

This chapter presents two extensions to the combined simulation and optimization framework presented in chapter 5. First, a two-phase evaluation approach is described in section 6.1. Then, we discuss another process for revealing the uncertain parameters in subsection 6.2. This process considers multistage decision making and aims to better imitate the real-world information structure.

6.1 Two-phase Evaluation Approach

The purpose of stage 2 is to evaluate the performance of the planned schedules generated in stage 1 under uncertainty. In order to generate a reliable estimate for the cost of uncertainty, many simulations may be required. However, some solutions clearly perform worse than others and may be discarded quickly. Specifically, solutions with many stock-outs will perform poorly due to the high cost associated with this event. In this section, we describe a two-phase extension of the framework presented in chapter 5. The goal of this extension is to use the available computational time more efficiently by spending less time evaluating poor solutions and more time getting reliable estimates of the performance of the good solutions.

The first phase uses the framework from chapter 5 to generate potential solutions and get a rough estimate of the solutions' performance under uncertainty. Only a small number of challenging scenarios are used in the evaluation in this phase to reduce computational time. Once stage 1 re-chooses a solution that has been tested before, the candidate generation phase terminates. A subset of potential

 58

solutions that performed well in the first evaluation, called the candidate solutions, passes on to the second phase of the framework. The second phase uses stage 2 of the original framework to do a more thorough evaluation of the candidate solutions. In this evaluation phase, the goal is to find the best solution among the candidate solutions by evaluating them on a large number of scenarios. For a detailed illustration, see Figure 6.1.

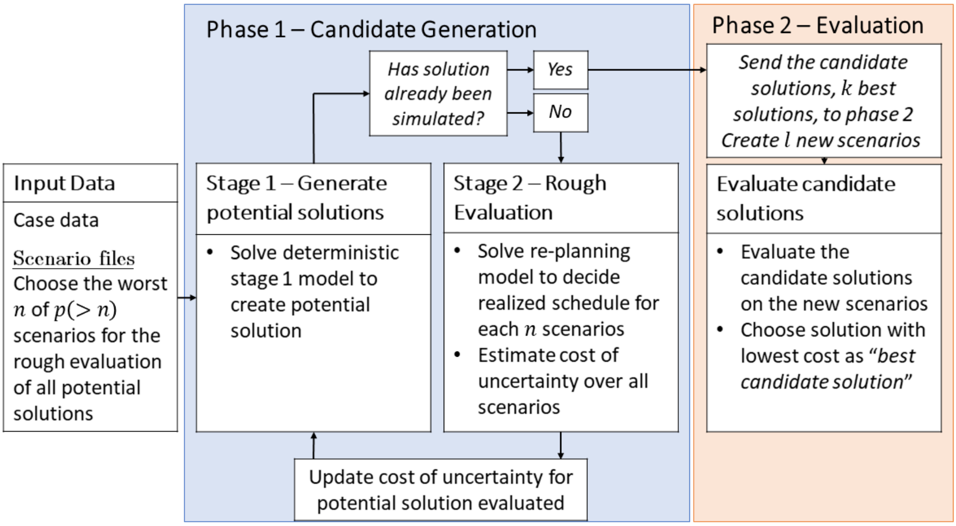

Figure 6.1: Illustration of the two part extension of the feedbackloop

As mentioned, the first phase of the framework is identical to the framework presented in chapter 5. However, there are two differences in how it is used. Firstly, the scenarios used to evaluate the potential solutions are different. Because infeasible scenarios have the greatest influence on the cost of uncertainty, the potential solutions are only tested on scenarios with a large probability of stock-outs. These challenging scenarios are chosen by generating p scenarios and choosing the $n < p$ scenarios with the greatest cumulative delay, delay summed over all ships, in the start time of operations in loading ports. Here we assume that the scenarios with the largest delays are the most likely to result in stock-outs. Furthermore, all potential solutions are evaluated over the same n scenarios. This increases the covariance between the performance of the solutions, thereby reduces the number of simulations needed to assess the difference in performance Fu (2002).

Secondly, phase 1 does not terminate with an optimal solution but rather with a set of k candidate solutions. These solutions performed best in the challenging conditions and we therefore assume that they will perform well in realistic conditions as well. This assumption rests on the fact that the cost of stock-out is high and influences the total cost to a large extent. However, there is no guarantee that the optimal solution will be a part of the set of k candidate solutions. Particularly, the candidate solutions may be too conservative if p is set too high. Therefore, the two-phase framework should be viewed as a heuristic approach where the objective is to find better solutions on average, but optimality cannot be guaranteed.

The second phase consist of evaluating the k candidate solutions generated in the first phase. Since the purpose is to find the solution performing best under uncertainty, the candidate solutions must be evaluated over a sufficiently large number of independent scenarios, l. This is done using the stage 2 re-routing model described in chapter 5. The scenarios should be equal for all candidate solutions due to the same argument as for the comparison of potential solutions in the first phase. After the evaluation process, the best performing candidate solution is selected. This solution is denoted "best candidate solution" in the rest of this thesis because we cannot guarantee optimality.

6.2 Multistage Solution Approach to Stage 2

So far, we have revealed all the uncertain parameters to the re-routing model in stage 2 simultaneously and solved the model for the entire planning horizon once for each scenario. However, in real life the information for the different ships is revealed at different times. Therefore, the information structure of the problem is not two-stage, but rather multistage in nature. In order to make the framework more realistic, we solve the re-routing problem in stage 2 in an iterative manner, moving forward in time and revealing some information with each iteration. This process is explained in further detail in subsections 6.2.1 and 6.2.2. The intention behind this method is to better capture the dynamics of the real-world decision process.

6.2.1 Step-Wise Release of Information

The key difference between the two-stage and the multistage approach is the timing at which information about the uncertain parameters is revealed. After a stage 1 solution has been produced, the uncertain parameters are the arrival times in the transit point T^s_{vf}. In the two-stage approach, we reveal this information for all the ships simultaneously and solve the stage 2 re-routing model once. Now however, the actual transit times are revealed at different times for different ships. This leads to a more realistic representation of the problem faced by Hydro because the stage 2 re-routing problem no longer has perfect information about the realized uncertain parameters for the entire period.

 60

The logic behind the approach outlined in subsection 6.2.2 rests on some assumptions about the uncertain parameters in the problem. We assume that the ships have no way of knowing at what time within the time interval they can commence loading in a loading port. However, we also assume that once a ship begins loading, it can use weather forecasts to correctly estimate the travel time to the transit node. Thus, the actual transit time should be available to all ships that have started loading or are currently sailing. Ships that have not yet started loading need to rely on the deterministic planned schedule. These assumptions represent our best attempt of emulating the real-world conditions faced by Hydro.

6.2.2 Stage 2 Iterative Evaluation Approach

In the iterative approach outlined in the pseudo code below, the evaluation of a potential solution from stage 1 is described. For each scenario $s \in \mathcal{S}$, the re-routing model can at most be solved as many times as there are ships. The transit point has a critical role in the process. We assume that this is the point where the final decision about a ships routing is taken. Once a ship arrives at the transit point, the amount of information that should be available about the arrival times of the other ships is determined.

Then, the routing decisions for the ship in the transit point are taken. Before the next iteration, we simulate a move forward in time until the next ship arrives at the transit point. The decisions from previous iterations are locked and the process is repeated until all ships have passed the transit point. Thus, we solve the re-routing model for the remaining time horizon in each iteration and the time horizon becomes progressively shorter. Note that the re-routing of the ships may be determined based on incomplete information about the delays of the later ships. This dynamic is key in imitating the real-world decision process.

Pseudo code for stage 2 with multistage information structure

(1) Input from stage 1 solution η: Set of used ships, $\mathcal{V}^2 \subset \mathcal{V}$, start time of loading, t_{im}, operational time in port, t^O_{imv}, and planned travel time to transit, T_{ifv}

(2) For $s \in \mathcal{S}$:

2.1 For $v \in \mathcal{V}^2$:

- Draw uncertain port delay τ^s_v and travel time to transit T^s_{ifv}
- Calculate actual start time of loading $T^s_{imv} = t_{im} + \tau^s_v$
- Calculate actual arrival time in transit $T^s_{fv} = T^s_{imv} + t^O_{imv} + T^s_{ifv}$

2.2 Sort ships $v \in \mathcal{V}^2$ based on T^s_{fv}

2.3 For iteration $\mu = 1...|\mathcal{V}^2|$

- Update current time to $t = T^s_{f\mu}$
- Update estimated arrival time in transit T^{μ}_{fv} for all ships:
  - $T^{\mu}_{fv} = T^s_{fv}$ for $v \in \mathcal{V}^2 : T^s_{imv} \leq t$
  - $T^{\mu}_{fv} = \max(t_{im} + t^O_{imv} + T_{ifv}, t + t^O_{imv} + T_{ifv})$ for $v \in \mathcal{V}^2 : T^s_{imv} \geq t$
- If T^{μ}_{fv} changed since previous iteration
  - Fix variables equal to value from previous iteration for $v \in \mathcal{V}^2 : v < \mu$
  - Run re-routing model
  - If infeasible: Break
  - Record new costs and schedule

2.4 Record costs in scenario, C^S:

- If feasible: $C^s =$ Realized cost
- If infeasible: $C^s =$ Planned cost + stock-out cost

(3) Calculate estimated cost of uncertainty for stage 1 solution η:
$\Delta C_{\eta} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} C^s -$ Planned costs

For a given scenario, the ships are first sorted in the order of actual visits to the transfer point. In each iteration μ, the current time t is set equal to the time at which the ship μ arrives at the transfer point. At this time, the actual arrival time in the transit point T^s_{fv} is revealed for the ships that have already begun loading in a port. If a ship has not yet started loading, the maximum of the earliest possible arrival time in transit and the planned arrival time in transit is used. The earliest possible transfer time is found by assuming that a ship starts loading immediately.

Then, the decision variables for all the ships that have already passed the transit point are fixed. We do this because we assume that the routing of a ship cannot be

 62

further changed during the visits to the unloading ports. Furthermore, we remove the possibility of hindsight. The re-routing model is then used to determine the unloading and routing decisions for the ship in the transit point and all the ships that have yet to reach the transit point. Finally, the process moves one iteration forward and repeats itself until all the ships have passed the transit point.

An illustration of the process is presented in Figure 6.2. In this figure, the approach is in iteration 2 where the second ship has just arrived at the transit node. Ships 3 and 4 have started loading in the loading port while ship 5 is delayed and waiting outside a loading port. Ship 6 is yet to be used. In this situation, the model will use the actual transit times for ships 1-4, the earliest possible arrival time in transit for ship 5 and the planned arrival time in transit for ship 6. The variables for ship 1 are locked because they were decided in iteration 1. For ships 2-6, the second stage routing and unloading variables are free.

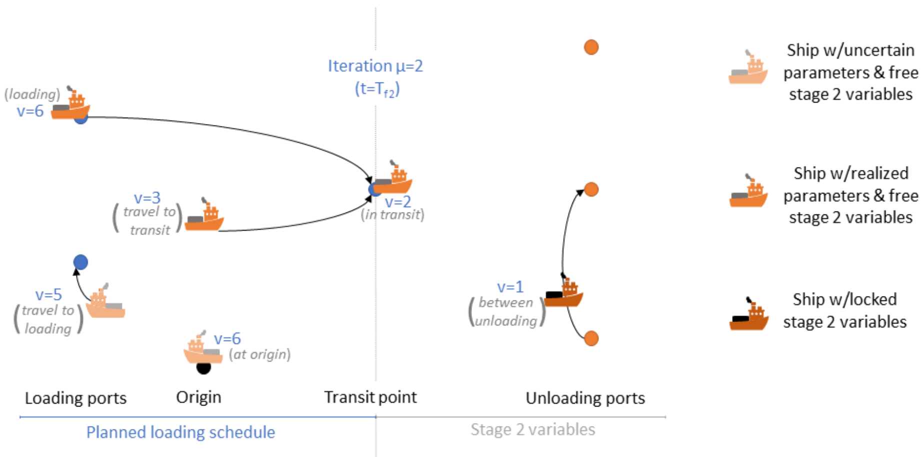

Figure 6.2: Illustration of one iteration in stage 2

In order to speed up the solution time, we want to solve the re-routing model as few times as possible. Therefore, we only solve the re-routing model if any new information has been revealed since the previous iteration. If the exact same arrival times in the transit point, T^s_{vf}, are used in two iterations, they will produce the exact same delivery schedule and there is no need for a re-routing.

Chapter 7

Case Study

This chapter provides an overview of the data used to test the solution framework. Since Hydro's data is sensitive, most of the data is gathered from publicly available sources. However, the data is gathered with the objective of mimicking Hydro's real-life problem. The first two sections 7.1-7.2 describe the unloading and loading ports, respectively. The characteristics of the ships are presented in section 7.3, while the cost data is described in section 7.4. Further, the uncertain parameters are discussed in section 7.5. Lastly, the time horizon used in this thesis is presented in section 7.6. Please note that sections 7.1-7.4 and 7.6 describe a realistic, deterministic problem instance for Hydro and are based on the data gathered in Nikolaisen and Vågen (2018).

7.1 Unloading Ports

The unloading ports in this thesis comprise of Hydro's Norwegian smelters. Each smelter has a given consumption rate, inventory limitations and unloading rate for each ship class. The latter is provided by Hydro, while the former two are estimated. A summary of the key data for the unloading ports is presented in Table 7.1.

Consumption Rate

In order to estimate the annual consumption of alumina, the plants' annual production capacity of primary aluminium is multiplied by 2. The capacity is found in the annual report (Hydro 2018), while the factor 2 is used because 2 kg of alumina is needed to produce 1 kg of aluminium (Hydro 2012). As discussed in subsection 2.2.3, the production rate at the smelters is stable and not affected by seasonality.

Therefore, the daily consumption rate of alumina is assumed to be deterministic and constant in this thesis.

Inventory Constraints

The minimum and maximum inventory levels are neither provided nor publicly available. Therefore, we make some assumptions that are subject to considerable uncertainty.

In the estimation of minimum level of inventory, we assume that each plant has a safety stock equal to 7 days of consumption. This is further rounded to the nearest 1000 tonnes. As the closest supplier is 3 to 4 days sailing away and loading can take 1 to 3 days, this safety stock will allow for the possibility of a rush order. Further, a higher minimum inventory level is imposed at the end of the planning period to avoid unfavourable end-of-horizon effects. We estimate this level as 15 days of consumption rounded to the nearest 1000 tonnes.

In practice, the storage capacity of the plants varies significantly depending on the historical investments that have been made. To capture this dynamic where some plants need frequent deliveries while others can have infrequent deliveries, the maximum inventory is estimated as a multiple of the daily consumption. The multiple assigned to the unloading ports ranges randomly between 20 to 60 days and the maximum inventory is rounded to the nearest 5000 tonnes. These estimates seam reasonable given that each unloading port typically receives 1 to 3 shipment per month and has a minimum inventory level that equals 7 days of consumption.

Finally, the initial inventory level at an unloading port is chosen as a random number between the ending inventory requirement and the maximum inventory level, rounded to the nearest 1000 tonnes. With this approach, some unloading ports will require a delivery before a ship has time to load in Brazil and travel to Norway. Therefore, there is a need for ships in transit at the start of the planning period.

Table 7.1: Input data for unloading ports (tonnes)

	Daily Rate		**Inventory Levels**			
Plant	**Consumption** $[R_i]$	**Unloading** $[L_i^T]$	**Min** $[\underline{S}_i]$	**Max** $[\overline{S}_i]$	**Initial** $[S_i^0]$	**End** $[\underline{S}_i^T]$
Karmøy	1 485	7 200	10 000	75 000	54 000	22 000
Årdal	1 090	7 200	8 000	45 000	32 000	16 000
Sunndal	2 230	6 480	16 000	55 000	37 000	33 000
Høyanger	362	5 280	3 000	20 000	7 000	5 000
Husnes	1 036	7 200	7 000	60 000	26 000	16 000

7.2 Loading Ports

In addition to Alunorte in Brazil, two other loading ports are considered in this thesis. They are chosen to offer a relevant South American supplier and a geographically close European supplier. Alumar in Brazil represent the former, as it is partially owned by Rio Tinto Alcan (Rio Tinto 2018), Hydro's main external alumina supplier alumina (Hydro 2018). Aughinish in Ireland is chosen as the latter, as it is the largest alumina plant in Europe.

The loading rate is chosen to be representative of the plant size but not necessarily accurate. Representative loading rates are provided by Hydro. Please note that although the optimization models presented in chapter 5 can handle multiple products, we only consider one product in this thesis. An overview of the loading ports and their parameters is presented in Table 7.2.

Interval for Supplied Amount of Alumina

Hydro's actual supply intervals for annual amount of alumina picked up at the loading ports are specified in long term contracts and are not publicly available. Therefore, we approximated the intervals by matching the total supply of alumina to the consumption in the unloading ports. The reasoning behind this method is that matching supply to demand is a logical way of planning for the contracts in the first place. Over the planning period, total supply is set equal to total consumption minus any load on a ship in transit at the beginning of the planning period. The total upper supply limit is then found by adding 15% contractual slack while this slack is subtracted to find the lower limit.

Alunorte is assigned the majority of the supply, 60%, because it is Hydro's main supplier of alumina. Alumar and Aughinish are each assigned 20%. These estimates are highly uncertain and represents a best guess. The supply limits are then rounded to match the capacities of the ship classes. This results in some deviations from the calculated assigned supply for each individual port, but maintains the total supply limits for all loading ports.

Table 7.2: Input data for loading ports (tonnes), 60 days planning horizon

Port	**Yearly capacity**	**Daily loading rate [L_i^T]**	**Supply limit Lower [$\underline{L}_i$]**	**Supply limit Upper [$\overline{L}_i$]**
1: Alunorte	6 400 000	25 000	150 000	200 000
2: Alumar	3 500 000	20 000	50 000	70 000
3: Aughinish	1 900 000	16 000	50 000	65 000

7.3 Ships

Ship Classes and Capabilities

Because we assume that Hydro is free to choose the ship class they prefer, the number of ships in each class must be large enough to not restrict the problem. Therefore, we seek to have as few ship classes as possible while covering the capacity range that Hydro uses. In this thesis we consider three ship classes. The characteristics of the ship classes are summarized in Table 7.3. Since we have not observed a well-defined ship class with less capacity than Handysize, a class with 15000 tonnes in capacity is introduced.

Since each ship needs to lower speed when traveling close to shore, the cruising speeds in Table 7.3 are somewhat conservative compared to the cruising speeds of an average Handysize ship (Wikipedia 2018).

Table 7.3: Input data for ship classes

Class	**Capacity (tonnes)**	**Cruising speed (knots)**
1: Small ships	15 000	11
2: Handysize	35 000	12
3: Handymax	50 000	13

Contracted Number of Shipments

As mentioned in section 2.3.1, the shipping contracts establish an interval for the total number of shipments Hydro has available in the time horizon. Similarly to the alumina supply contracts, the shipping contracts are approximated by first estimating the required number of shipments based on consumption at the unloading ports and an assumed average ship size of 40 000 tonnes. Then, the upper limit is created by adding slack of 15% and rounding the figure up while the lower limit subtracts the slack and rounds down. This average capacity is used because Handysize and Handymax are the most commonly used ship classes in practice. The result is an interval of $[8, 11]$ shipments over the 60 days time horizon.

Initial Location of Ships

To avoid forced use of the loading port in Ireland or stock-outs in the beginning of the planning horizon, the system needs some ships in transit at the start of the planning period. The travel times between South America and Norway are approximately 20 days. Further, by considering the initial inventory levels and the consumption rates, the three unloading ports will reach minimum inventory limit within 20 days. Therefore, two full ships, one of class 2 and one of class 3, are assumed to be in transit. The first is one and a half days away from the transit point, while the second is one week behind the first. Note that the routing of the transit ships is not fixed, they are simply closer to the unloading ports.

7.4 Costs

Transportation Costs

We assume that the daily operating costs for a ship consist of two major components; time charter rate and bunker cost. The time charter rate (TCE rate) is estimated from 2017's average bulk carrier TCE rates provided by Clarksons Platou (Clarksons Platou 2018). Handysize and Handymax rates are provided directly while the cost for class 1 is estimated by multiplying the rate for Handysize with a factor of 0.7. This factor is larger than the relative capacities of the two classes would suggest because hire rates per tonne usually decrease as ship capacity increases.

Daily bunker costs are approximated by multiplying an estimated daily bunker consumption (Doskocz 2012) with a quote for bunker price (Ship and Bunker 2018). A currency rate of 8.3 is used when converting the dollar denominated costs to Norwegian Kroner.

To calculate the cost of transportation for a ship, we use the deterministic travel times calculated as the distance between the ports divided by the sailing speeds in Table 7.3. Due to the design of the solution framework, each ship travels through the same geographical point in the North Sea, called the transit point. The distance between loading and unloading ports in the stage 1 problem is calculated as the distance between the loading port and the transit point plus distance between the transit point and the unloading port, taken from Sea-distances.org (2018). In this thesis, we use the port of Aberdeen as the transit point due to its suitable geographical position. Introducing Aberdeen as a transit point leads to a maximum error if the travel times between loading and unloading ports of 0.7 days. We consider this inaccuracy acceptable.

The cost of hiring the ships during operation is calculated as the daily transport cost multiplied with the operating time for each visit. The operational time for a visit consists of a fixed setup time in the port and the (un)loading time. Even though the model allows the setup time to be dependent on both ship class and port, we assume it is 0.5 days for all ship classes in each port.

Fixed Port Costs

In addition to the transportation costs, there are fixed costs associated with entering a port in both loading and unloading ports. In practice, these port fees are unique for each port. However, public information about the ports in question in this thesis is not available. Therefore, the cargo due costs for the port in Bergen are used as the cost for all ports (Port of Bergen 2018). We use the cargo dues because they were the dominant component of the port costs and Bergen because they quote the cargo dues for bulk products that resemble alumina. Furthermore, other Norwegian ports quoted similar dues for other products. A summary of all ship costs is presented in Table 7.4.

Table 7.4: Cost data for ship classes

Class	**TCE rate**	**Bunker consumption (Tonnes / day)**	**Transportation cost [c_k^T] (NOK / day)**	**Port cost [c_k^P] (NOK / visit)**
1	57 870	18.0	132 570	135 000
2	82 671	22.5	176 046	315 000
3	102 159	27.0	214 290	450 000

Cost of Inventory Stock Out

When running the stage 2 problem there is a chance of reaching infeasiblity with the suggested stage 1 solution. As discussed in 5.2, we choose to handle infeasible stage 2 solutions by assigning a stock-out cost to the simulation. Infeasible stage 2 solutions will only arise because the inventory drops below the safety stock in a port. Because each port has 7 days worth of consumption in safety stock, this will not cause a plant to immediately stop production. Therefore, we do not set an infinitely high penalty cost for unfeasible solutions, but rather use a penalty cost estimated as the cost of a rush order of alumina. This cost should be high enough to make infeasibility highly undesirable, but not so high that a single stock-out excludes a stage 1 solution from consideration.

We assume rush orders happens with a small ship from the loading port in closest proximity, Aughinish. The cost of renting a ship for 10 days and paying the port costs equals around NOK 2 mill. We further assume that the 15000 tonnes of alumina needs to be bought at a 20% premium to market price, a premium of around 100$ per tonne. This adds a further NOK 13 million to the penalty cost. To account for the possibility that a rush order will not be possible and the plant will have to stop production, we include an additional NOK 5 mill., bringing the total penalty cost for an infeasible solution up to NOK 20 millions.

7.5 Uncertain Parameters

Uncertainty in Start Time for Loading

As discussed in section 2.4, the time that a ship begins loading in a loading port is uncertain. In reality, Hydro gets a time window from the loading ports. Within this time window, the ship can be admitted into the port at any time. According to Hydro, it is reasonable to assume the arrival time as uniformly distributed across the time window. The length of the time window for each loading ports is given in Table 7.5.

Table 7.5: Input data for length of time windows in loading ports

Port	Length of time window (days)
1: Alunorte	± 3.5
2: Alumar	± 3.5
3: Aughinish	± 3.5

Uncertainty in Travel Time

Due to weather conditions the ships can experience delays in travel time. To incorporate it into the system, we assume that the uncertain delay occurs on the path between the loading ports and the transit node. Further, we assume that the possible delay is proportional to the expected travel time. Therefore, the expected travel time is multiplied with a weather dependent factor.

In this thesis, we choose to model the weather in a simple way. The weather can either be good, decent or bad. These three outcomes are equally likely. When a ship travels from South America to Norway, Hydro has historically experienced a change from planned to actual time of arrival between ± 2 days. This equals around $\pm 15\%$ in actual travel time. The distribution used in this thesis is presented in Table 7.6.

Table 7.6: Input data for delay in travel times

Weather condition	Probability	Time factor
Bad	1/3	1.20
Decent	1/3	1.05
Good	1/3	0.90

7.6 Time Horizon

The stage 1 planning problem is of a tactical nature and Hydro's planning division currently makes delivery schedules for the next one to six months. However, the complexity of the problem increases considerably when we include uncertainty and evaluate the stage 1 solution with the operational flexibility of re-routing in stage 2. Therefore, somewhat shorter time horizons of 45-60 days are considered in this thesis. The data presented in the chapter are based on a planning horizon of 60 days.

Chapter 8

Computational Study

This chapter presents the results from the computational study of the solution framework. Section 8.1 describes the details regarding problem instances. The performance of the solution framework is presented in section 8.2. Then, the results of our efforts to reduce the runtime in stage 1 and stage 2 are described in sections 8.3 and 8.4, respectively. Thereafter, we present the results of the modified simulation approach in 8.5 before discussing the results of the multistage extension in 8.6. Finally, the complete framework is analysed in 8.7.

The optimization models are written in Mosel and implemented in FICO Xpress. The Solution Framework including the simulation and two-phase evaluation approach are written and implemented in Python. All tests have been run on a computer with processor 2x Intel Xeon Gold 5115 with 10 cores and 96Gb RAM and CPU 2.40GHz. We have used the optimization software Xpress version 8.5.10 and the Python version 3.7.2-foss-2019a compiled with gcc6.4.0 (GNU Compiler Collection). The optimality gap in Xpress is set to 1% for both the stage 1 and the stage 2 optimization model.

8.1 Problem Instances

The data used to mimic Hydro's planning problem is described in chapter 7. In order to test the solution framework, we introduce two realistic problem instances with varying time horizon. As the time horizon increases, the number of ships also increases. These instances are denoted *Medium* and *Large*. Further, we include a small instance, *Small*, to illustrate the workings of the framework within a shorter solution time. A summary of the size of all instances used in the computational study is presented in Table 8.1.

Table 8.1: Problem Instances

	|T|	|P|		|K|	|V|
Instance	**Time periods**	**Loading ports**	**Unloading ports**	**Ship classes**	**Ships (total)**
Small	45	1	3	1	8
Medium	45	3	5	3	18
Large	60	3	5	3	24

8.2 Basic Solution Framework

In this section, we seek to evaluate the solution framework without any extensions. First, how quickly the framework converges towards an estimated optimal solution is discussed in subsection 8.2.1. Then, the quality of the chosen solution is evaluated in subsection 8.2.2.

8.2.1 Speed of Convergence

In each iteration of the framework, the stage 1 model is solved once and the stage 2 re-routing model is solved $|S|$ times as a single solution is found and evaluated. Therefore, the number of solutions that needs to be evaluated before an optimal solution is found, N, greatly influences the total runtime of the framework. Table 8.2 presents a summary of the runtime for the framework for the two smallest problem instances. The *Large* instance does not converge within a reasonable runtime. Hence, it is not included in this first part of the computational study.

Table 8.2: Average runtime for 5 runs of the framework, $|S| = 20$

Instance	ΔT	N	Total time [s]	% time in stage 1
Small	**5**	14	987	36 %
	3	71	4 741	41 %
	1.5	161	10 905	46 %
Medium	**5**	36	14 071	34 %
	3	109	41 599	34 %
	1.5	> 200	> 66 000	36%

The first thing to note is that the speed of convergence is highly dependent on the length of the time interval, ΔT. As discussed in subsection 5.5.3, we discretize the time horizon into intervals of length ΔT in order to compare solutions. As ΔT

decreases, the number of unique stage 1 solutions increases. In fact, the number of solutions that needs to be tested, N, more than doubles when we halve ΔT. This poses a challenge because we ideally want ΔT to be as small as possible. To determine a reasonable ΔT, we need to look at the trade-off between runtime and the risk of missing good solutions. Secondly, although the framework converges towards an optimal solution, it does not converge quickly enough to solve the *Medium* instance for $\Delta T = 1.5$.

Further, approximately one third of the time spent generating potential solutions in stage 1 and two thirds is spent evaluating potential solutions in stage 2. In order to reduce the runtime of the framework, we therefore work on reducing both the runtime of stage 1 and of stage 2 in the coming sections.

8.2.2 Solution Quality

In this computational study, the performance of a solution mainly refers to the estimated cost of a stage 1 solution. This cost is determined in the evaluation of a solution and is the sum of the real costs and the penalty cost imposed on scenarios with stock-outs. Therefore, a stage 1 solution with good quality has a low risk of stock-outs in stage 2 and low real costs. Please note that infeasible scenarios and scenarios with stock-outs will be used interchangeably and refers to the same thing.

The quality of the estimated optimal solution chosen by the framework is sensitive to both the choice of ΔT and the number of simulations $|S|$. If the number of simulations $|S|$ is too small, the cost of uncertainty will not be reliable. This is particularly true when the cost of uncertainty is sensitive to low probability events. In our case the cost of infeasibility is high and a low percentage of stock-outs will impact the cost of uncertainty to a large degree. Further, the framework terminates when the stage 1 model re-chooses a stage 1 solution that has been tested before. If ΔT is too large, the best solutions might be overlooked because it is deemed to be equal to a previously tested solution.

Ideally, we would like to have the ability to use large number of simulations and a very small ΔT. However, both these parameters influence the solution time negatively. Therefore, there is a trade-off to consider. In order to evaluate the initial setting of $|S| = 20$, we need to check the performance of the estimated optimal solution. To do this, a re-evaluation is introduced where the estimated optimal solution is tested on 50 new scenarios. A summary of the results for the *Small* instance can be found in Table 8.3. The costs are presented as a percentage of the deterministic optimal cost.

Table 8.3: Average solution quality for 5 runs of the *Small* instance, $|\mathcal{S}| = 20$ in solution framework, $|\mathcal{S}| = 50$ in re-evaluation

ΔT	**Estimated cost [% of optimal]**	**Re-evaluated cost [% of optimal]**	**Percentage change**
5	100.41 %	101.9 %	1.5 %
3	100.27 %	101.0 %	0.7 %
1.5	100.18 %	102.6 %	2.4 %

As expected, the estimated cost decreases when ΔT decreases. Choosing a low ΔT allows the framework to consider more unique stage 1 solutions, thereby it can terminate with a better estimated optimal solution than for larger values of ΔT. However, the difference in estimated cost between $\Delta T = 5$ and $\Delta T = 3$ is only 0.14%. The difference between $\Delta T = 3$ and $\Delta T = 1.5$ is even smaller with a drop of 0.09%. This marginal improvement may not be worth the extra computational time for larger problem instances and we choose to continue with $\Delta T = 3$ for the rest of the computational study.

Table 8.3 further shows that the cost of the optimal solution increases in the re-evaluation for all values of ΔT. This increase is mainly driven by instances of stock-outs in the re-evaluation, approximately 2% of the scenarios, that did not occur in the first evaluation. Although the average increase in cost of around 1.5% is not dramatic, we believe it is not acceptable. Thus, we conclude that 20 simulations is not quite enough to get an accurate estimate of a solutions performance.

8.3 Stage 1 Performance

To reduce the stage 1 solution time, there are broadly speaking two courses of action. The time it takes to solve the stage 1 optimization model can be reduced and the number of times it needs to be solved can be reduced. In this section we first present improvements to the optimization model in 8.3.1. Then we discuss how to reduce the number of time the model is run by extracting multiple candidate solutions in each run in 8.3.2.

8.3.1 Stage 1 Optimization Model

The stage 1 optimization model represents a further development of the continuous time formulation and valid inequalities developed in Nikolaisen and Vågen (2018). In this subsection we will present the improvements made and the performance of the model before discussing how the number of variables can be reduced after the first iteration of the framework.

Reducing Solution Time by Valid Inequalities

The valid inequalities presented in chapter 5 are summarized in Table 8.4. Knapsack inequalities and subtour restrictions were also tested, but they performed to poorly to be included in the computational study.

Table 8.4: Summary of the Valid Inequalities Tested

Valid Inequality abbreviation	Description
Non	Run without any valid inequalities
t^O_{imv} & $\underline{M}_i$	Valid inequalities for total operational time in each port and minimum number of visits
T^{MAX}_{im}	Valid inequalities for upper limit on time visited variables for each visit in each port

The valid inequalities presented have been tested on both the *Medium* and *Large* instances, both alone and in combination. The results are presented in Table 8.5.

Table 8.5: Stage 1 runtime in seconds and percentage of *Non*

Valid inequality	*Medium*	*Large*
Non	283	1 293
t^O_{imv} & $\underline{M}_i$	459 (+62%)	1 120 (-13%)
T^{MAX}_{im}	206 (-27%)	3 470 (+168%)
t^O_{imv} & $\underline{M}_i$ + T^{MAX}_{im}	273 (-3%)	905 (-30%)

Both instances experienced reduction in runtime when both t^O_{imv} & $\underline{M}_i$ and T^{MAX}_{im} where implemented, while the effect of implementing a single inequality was inconclusive. Therefore, we choose to implement t^O_{imv} & $\underline{M}_i$ and T^{MAX}_{im} in both the stage 1 and stage 2 models.

Constraining the Solution Space of the Stage 1 Model

The number of variables in the model depends on the set of ports, the set of possible visits and the set of ships. In the first iteration of the framework, both the set of possible visits and the set of ships needs to be sufficiently large to not restrict the problem. However, after the stage 1 has been solved once, we get an idea of what a good solution may look like. We attempt to utilize this information to reduce the size of the set of ships and possible visits.

Let $\text{MaxPreviousVisits}_i$ define the maximum visits in port $i \in \mathcal{P}$ in any of the previous iterations of the framework. Further, let $\text{MaxPreviousShips}_k$ define the maximum number of ships from class $k \in \mathcal{K}$ used in any of the previous iterations. Finally, we introduce the parameters ExtraShips and ExtraVisits and redefine the set of possible visits and ships in a class as follows.

$$\mathcal{M}_i = \{1, ..., \text{MaxPreviousVisits}_i + \text{ExtraVisits}\}$$
$$\mathcal{V}^k = \{1, ..., \text{MaxPreviousShips}_k + \text{ExtraShips}\}$$

When testing the framework without constraining the solution space, the difference between the solution with the most ships in a class and the solution with the least ships in a class was 2. Similarly, the maximum difference in planned visits was also 2. Therefore, we set both ExtraShips and ExtraVisits equal to 2 for the rest of the computational study in order to have a low risk of excluding the best solutions. That means that there are always two more ships in a class and two more visits in a port available than has been used in any of the previous iterations. The effect of redefining the sets on model runtime is presented in Table 8.6.

Table 8.6: Stage 1 runtime development in feedback loop [s]

Iteration Nr.	*Medium*	*Large*
0	326	1 419
1 - 20	119	1 281
20 -	125	1 821

By reducing the model size after the initial iteration, the solution time is reduced considerably to about 40% of initial runtime for the *Medium* instance. This is due to the constraining of the solution space, a heuristic approach to reduce runtime. For the *Large* instance, the effect is more modest. Secondly, there is an incremental increase in the runtime as the feedback loop progresses. The stage 1 model increases in size for each iteration, due to the additional constraints to connect the correct penalty cost to solutions tested in previous iterations. While this effect is small for the *Medium* instance, it is more pronounced for the *Large* instance.

8.3.2 Extracting Multiple Potential Solutions in Stage 1

The goal of extracting multiple potential solutions in stage 1 is to reduce the total number of times the optimization model needs to be solved. However, we only want to extract solutions that are close to optimal in order to avoid spending time evaluating poor solutions. We further don't want to extract solutions that have already been tested. In order to achieve this flexibility, a call-back function in Xpress is used to test each integer solution found. If a given stage 1 solution has an optimality gap of less than 5% and has not been tested before, it is added to the set of solutions to be extracted. If the maximum number of solutions to be extracted has been reached, the new solution overwrites the poorest solutions in the set. This way, the N stage 1 solutions with the lowest planned costs are stored. Note that the model is still solved to optimality in each iteration.

With the standard settings, Xpress often finds less than 3 integer solutions for our instances. In order to extract more integer solution, some setting are changed compared to section 8.2. Firstly, the node search strategy is changed to depth first. Secondly, the cut-off value in the branch and bound tree is changed such that non-improving solutions are accepted until the maximum number N has been reached. Thirdly, the effort spent on heuristic searches is increased by a factor of 3.

The average results over 5 runs of the framework is presented in Table 8.7. The "Max extracted solutions" parameter determines how many candidate solutions we allow to be stored in each iteration of stage 1. Increasing the parameter to more than 40 occasionally crashed the computer. Therefore, the analysis stops at 40 even though the runtime appears to continue to decrease.

Table 8.7: Average stage 1 data when varying the parameter controlling maximum number of solutions possible to extract from stage 1, $\Delta T = 3$, *Medium* instance

Max extracted solutions, N	**Avg. solutions extracted per iteration**	**Stage 1 solutions evaluated**	**Total runtime stage 1 [s]**
1	1	109	13 758
10	5	158	4 201
20	11	223	3 309
30	13	187	2 404
40	14	174	2 213

Table 8.7 clearly show that the stage 1 runtime reduces drastically as more solutions are extracted from each iteration. As the average number of solutions extracted increases from 1 to 14, the time spent on stage 1 reduces by a factor of 6. There are two reasons why the decrease in runtime is not proportional to the increase in the number of solutions extracted. When sub-optimal stage 1 solutions are extracted, the average number of solutions that need to be tested before the framework terminates increases. Furthermore, the solution time for each iteration of the stage

1 model increases when the Xpress settings are changed.

8.4 Stage 2 Performance

Stage 2 accounts for around two thirds of the framework's runtime. In order to reduce the solution time, we propose three courses of action. Firstly, we examine the performance of the optimization model in subsection 8.4.1. Then, we present the results of evaluating the potential solutions in parallel in subsection 8.4.2. Finally, we attempt to reduce the number of times the stage 2 model is run by stopping the evaluation of poor solutions early in subsection 8.4.3.

8.4.1 Stage 2 Optimization Model

The stage 2 re-routing model needs to be solved $|\mathcal{S}|$ times for each time we solve the stage 1 model. Furthermore, the stage 2 model controls how much flexibility we allow when re-routing the ships after the uncertain parameters have been realized. In this subsection, we first present the solution time of the model. Then, we discuss how to control the flexibility of the framework.

Solution Time

A comparison of the solution time for the stage 1 and stage 2 models is presented in Table 8.8. Note that the stage 2 re-routing model is significantly quicker than the stage 1 model and can be solved to optimality for all instances within a reasonably short time. In fact, the stage 2 model is solved approximately 10-14 times faster than the stage 1 model.

Table 8.8: Comparison of average runtime [s] per solution in stage 1 and stage 2, ΔVisits $= 2$

Instance	Stage 1	Stage 2
Small	27.5	2.0
Medium	126.0	12.7
Large	1 633.0	129.4

The improved efficiency is due to a few factors. Firstly, the network is smaller because we only consider unloading. Secondly, only ships that are used are included in the set $\mathcal{V}^2$ in the re-routing model. Therefore $\mathcal{V}^2$ is much smaller than $\mathcal{V}$ in stage 1. Thirdly, restricting the flexibility in stage 2 has a large impact on the solution time.

Limiting the Re-Routing Flexibility in Stage 2

If no further restriction is added to the stage 2 re-routing problem, the model can re-route the ships and change the unloading quantities however it sees fit compared to the planned schedule from stage 1. Thus, there is flexibility in both the unloading and routing decisions. However, large and frequent changes to the schedule may be impractical. In practice it may therefore be desirable to limit the flexibility in stage 2. The flexibility in unloading and routing can be limited in many ways. In this thesis, we choose to impose a limit to the re-routing flexibility by restraining the total number of visits to a given unloading port over the entire time horizon. By forcing the number of visits to not deviate from the plan by more than ΔVisits, much of the routing flexibility and all of the unloading flexibility is retained while each port maintains some degree of predictability in their operations.

In terms of implementation, the restrictions presented in constraints (8.1) are added to the stage 2 re-routing model. M_i^A represents the planned number of visits in port i from a given stage 1 solution. The set $\mathcal{M}_i$ are the potential visits to port i and $y_{im} = 1$ if a ship makes visits number m to port i.

$$\max\left\{M_i^A - \Delta\text{Visits}, 0\right\} \leq \sum_{m \in \mathcal{M}_i} y_{im} \leq \min\left\{M_i^A + \Delta\text{Visits}, |\mathcal{M}_i|\right\}, \qquad i \in \mathcal{P}^D \tag{8.1}$$

By altering ΔVisits, we can thus control the degree of flexibility. Choosing a low value for ΔVisits will also restrict the problem and reduce the solution time. The effect of varying ΔVisits in stage 2 on the average number of scenarios that are infeasible and runtime for the *Medium* instance is illustrated in Table 8.9.

Table 8.9: Effect of varying the parameter ΔVisits on the behaviour of stage 2 for the $Medium$ instance

ΔVisits	**% infeasible scenarios**	**Avg. Stage 2 model runtime [s]**
0	76 %	0.2
1	7 %	4.3
2	6 %	12.7
3	5 %	27.1

Table 8.9 shows that increasing ΔVisits from 0 initially has a huge effect on reducing the percentage of infeasible solutions in stage 2. Increases past ΔVisits equal 1 has a more modest effect. Further, the runtime of the stage 2 model more than

doubles each time ΔVisits is increased. We initially choose ΔVisits equal 2 since we want to have some re-routing flexibility in the system, while still being able to solve the simulation framework in a reasonable amount of time. However, we pick up the analysis of flexibility in stage 2 towards the end of this chapter to explore the impact of ΔVisits on the estimated costs.

8.4.2 Parallel Evaluation in Stage 2

To further reduce the runtime, we would like to exploit the fact that modern computers often have multiple cores available. Stage 2 is well suited to evaluate different candidate solutions in parallel because the simulations are independent of each other. If enough computers were available, stage 2 could in principle be processed entirely in parallel such that each core solves a single scenario for a single candidate solution. This approach would require many computers but could in theory reduce the stage 2 solution time to the time of solving the stage 2 optimization model a single time.

In this thesis, we only consider using one computer with 10 cores at a time. Furthermore, we evaluate each separate stage 1 solution on a different core. Thus, the approach needs to be viewed in connection with extracting multiple stage 1 solutions. If 10 or more solutions are extracted for a given iteration of stage 1, 10 stage 2 problems are solved in parallel. An alternative approach could be to simulate different scenarios in parallel for one stage 1 solution. Table 8.10 present the effect of parallel processing on the stage 2 solution time.

Table 8.10: Average stage 2 data when varying the parameter controlling maximum number of solutions possible to extract from stage 1, $\Delta T = 3$, *Medium* instance

Max extracted solutions, N	**Avg. solutions extracted per iteration**	**Time stage 2 [s]**	**Time stage 2 per solution evaluated [s]**
1	1	27 733	254
10	5	18 678	118
20	11	17 039	76
30	13	11 643	62
40	14	9 122	53

Table 8.7 proves that parallel processing reduces the stage 2 runtime significantly. Even though the computers used in this thesis have 10 cores, the runtime is only reduced by a factor of 3. There are several attributes of the framework contributing to this behaviour and we would like to highlight two of them. If 10 solutions are

evaluated on a scenario simultaneously, the framework will only proceed to the next scenario when the evaluations of all 10 solutions are completed. Therefore, the time in stage 2 is determined by the maximum time used to solve the stage 2 model. This number is higher than the average time and largely explains the performance. The second attribute is the fact that some iterations yield 40 solutions while others only yield 1, decreasing the benefit from parallel processing.

8.4.3 Cut-off to Discard Poor Solutions

Stage 1 solutions that often experience stock-outs in stage 2 perform poorly under uncertainty. We want to utilize this fact by discarding these stage 1 solutions from consideration before the full number of simulations are completed. Therefore, we introduce a cut-off limit representing the maximum number of infeasible solutions in stage 2. If this limit is exceeded during the simulation process for a solution, no further scenarios are simulated for the given solution, thereby reducing stage 2 run time.

To take full advantage of the cut-off limit we would like to begin simulating the scenarios with highest probability of generating infeasible solutions. As discussed in 7.5, a solution is infeasible due to stock-out in an unloading port. This will only happen when a ship arrives later than planned in an unloading port. We assume that the scenarios with the greatest cumulative delay, delay summed over all ships, are the most likely to result in stock-outs. Since the greatest factor of uncertainty is the start time for operation in a loading port, we use this when calculating cumulative delay. Thus, we sort the scenarios after cumulative delay and begin with the scenario with the largest delay.

Combining scenario sorting and cut-off limit leads to fewer scenarios simulated in total since the evaluation is sometimes stopped after few simulations. Remark that this change does not impact the solutions that perform well but reduces the time spent evaluating poor solutions. A summary of the results of varying the cut-off limit for number of stock-outs can be found in Table 8.11.

Table 8.11: Impact of cut-off limit on stage 2 solution time in *Medium* instance, $|\mathcal{S}| = 20$

Cut-off	Avg. Simulations per solution
20	20.0
2	18.0
0	14.6

The time spent in stage 2 is proportionate to the average number of simulations per potential solution. Therefore, we want to set a low cut-off limit in order to reduce the solution time. However, setting the cut-off to 0 seems too strict in our opinion because it excludes any potential solution with a single stock-out in stage 2. In general, we cannot be sure the optimal solution has zero stock-outs. Accordingly, we choose a cut-off limit of 2 when using 20 simulations, tolerating a maximum of 10% infeasible solutions in the evaluation.

8.5 Two-phase Evaluation Approach

This section presents the results of the two-phase extension to the simulation procedure detailed in subsection 6.1. First, the solution quality is explored in subsection 8.5.1. Thereafter, the runtime of the two-phase approach is presented in subsection 8.5.2.

As mentioned in section 6.1, the extension is an heuristic approach that evaluates the stage 1 solutions in two phases. First a rough evaluation is performed to narrow the set of potential solutions down to a small set of candidate solutions. Then, a thorough evaluation of the candidate solutions is performed to determine which is best. The goal of the extension is to improve the average cost and robustness of the stage 1 solution that the framework considers to be optimal without increasing the runtime significantly. In this computational study we have used scenarios as explained in Figure 8.1.

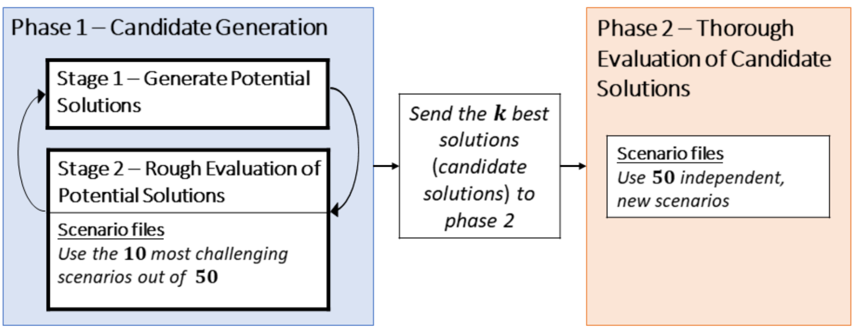

Figure 8.1: Explanation of the two-phase evaluation approach used

8.5.1 Solution Quality

In subsection 8.2.2, we conclude that 20 simulations are not quite enough to get an accurate estimate of the cost of uncertainty. Therefore, the estimated optimal solution does not perform as well as expected when evaluated again. The main reason for this is the large impact of stock-out on the estimated cost. In our opinion, there are two key questions to answer in order to determine if the approach works.

1. Does the evaluation of potential solutions on the 10 scenarios with the largest delay provide a set of robust candidate solutions?
2. How large does the set of candidate solutions need to be in order to be reasonably certain that it contains the best solution?

To test the first question, we ran the framework 5 times on the *Medium* instance with $\Delta T = 3$ and $k = 10$. Only 6% of the candidate solutions had any stock-outs during the thorough evaluation. As a comparison, around 15% of the estimated optimal solutions from the original framework resulted in one or more stock-outs when evaluated again. Thus, we conclude that the two-phase approach generates a pool of candidate solutions that are sufficiently robust.

In order to answer the second question, we saved a large number of candidate solutions, k, in the 5 runs of the framework. Figure 8.2 presents a box plot of the results from the thorough evaluation, where the boxes represent the results from the best $0-10$, $10-20$ and $20-30$ solutions from phase 1, from left to right. Note that the estimated costs are given as a percentage of the optimal deterministic costs for ease of comparison.

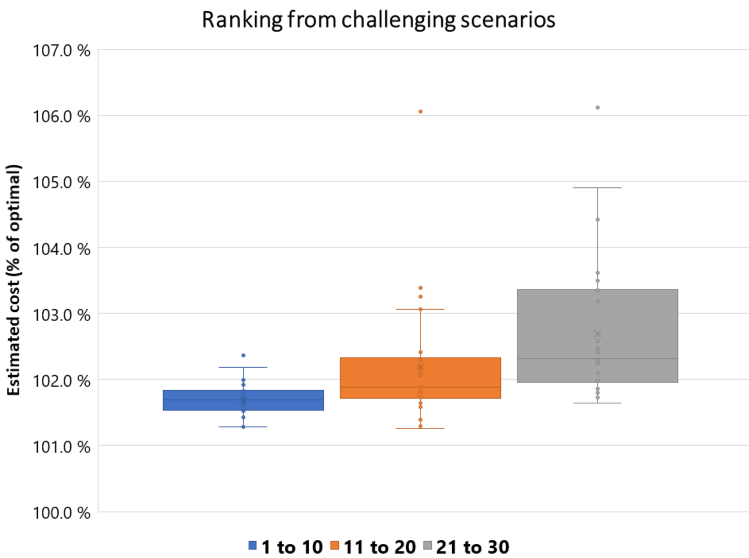

Figure 8.2: Box plot of estimated costs including stock-out penalty on 50 realistic scenarios sorted by evaluation on scenarios with large delays for the *Medium* instance

In this plot, the $0-10$ best candidate solutions tend to perform better than the $10-20$ in the thorough evaluation. Moreover, the best solution was found among the top 10 candidate solutions in all but 1 run. In this run, the difference in estimated cost was 0.02%.

Further, it is interesting to compare the estimated costs to the original framework. The $0-10$ best candidate solutions have an average estimated cost of 101.7% of the deterministic optimal cost. For the chosen optimal solution in the original framework, this number is 101.8%. Thus, the $0 - 10$ best candidate solutions perform marginally better than the solutions deemed optimal by the original framework.

Based on this data, we believe that the two-phase approach generates a better evaluation of the potential solutions than simply testing them on 20 random scenarios. The candidate solutions that remain after the first part are more robust than the estimated optimal solutions from the original framework. Furthermore, we can be quite certain that the best solution exists among the top 10 candidate solutions from the rough evaluation.

8.5.2 Solution Time

The second objective of the two-phase evaluation approach is to reduce the number of times the stage 2 model needs to run. The two-phase approach only uses 10 scenarios to test a stage 1 solution in the rough evaluation as opposed to 20 in the original framework. However, the thorough evaluation requires 50 additional simulations for each candidate solution, offsetting some of the gains.

Figure 8.3 summarizes the result of the work we have done to reduce the runtime of the framework. The two-part simulation approach with cut-off reduced the stage 2 solution time by a further 40% compared to the original framework with parallel processing and multiple solutions extracted. In total, the runtime has been reduced by more than 80%.

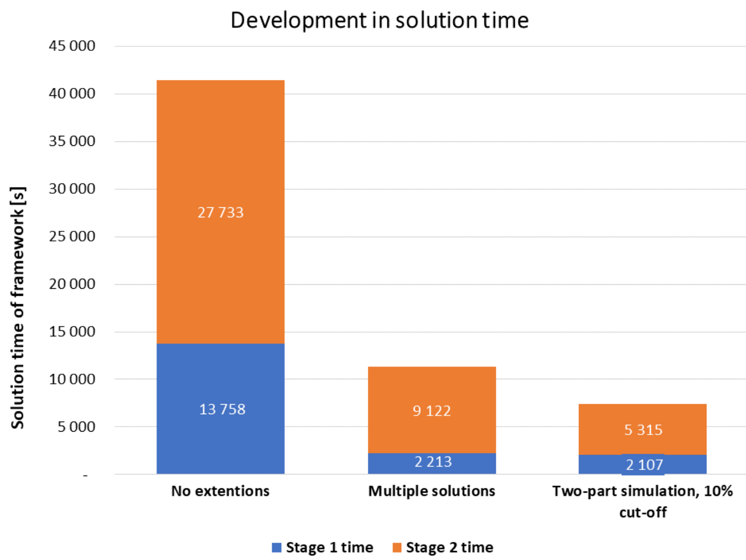

Figure 8.3: Impact of extensions on solution time for the *Medium* instance, $\Delta T = 3$

8.6 Multistage Information Structure

In order to make the framework more realistic, we introduced a multistage extension to the stage 2 re-routing problem in section 6.2. This section begins with an examination of the behaviour of the new model for evaluation in subsection 8.6.1. Then, the solution time is discussed in subsection 8.6.2.

8.6.1 Solution Behaviour

The objective of the multistage evaluation in stage 2 is to capture the challenges related to planning under incomplete information. Stage 1 solutions that appear robust when all uncertainty is revealed at once may perform differently when only partial information is available in the re-routing phase. In this subsection, we compare the behaviour of the framework when using two-stage information structure and multistage information structure in the evaluation. The goal is not to compare the solutions directly, but rather to explore if the evaluation changes materially.

A summary of the behaviour of the two-stage and multistage models for the *Medium* instance averaged over 5 runs is presented in Table 8.12. All runs are performed with the two-phase evaluation approach. Column 2 and 3 present the number of stage 1 solutions that have one or more stock-outs in the rough and thorough evaluation, respectively. Column 4 presents the estimated cost, including penalty for stock-outs, for the best candidate solution in the thorough evaluation.

Table 8.12: Comparison of behaviour of two-stage and multistage evaluation, $\Delta T = 3$, *Medium* instance, average of 5 runs, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Stage 2 model	Potential sol. with stock-outs	Candidate sol. with stock-outs	Cost of best solution [NOK]
Two-stage	41 %	6 %	27 816 066
Multistage	81 %	36 %	27 905 020

The first takeaway from Table 8.12 is that the average potential solution performs much worse when evaluated in the multistage model. This is because the two-stage model releases all the information simultaneously, making it easier to re-plan for delays that have yet to happen. In reality, this is not possible, and it represents a situation with perfect information in the re-routing phase. Further, among the set of candidate solutions, one third had one or more stock-outs in the thorough evaluation. Finally, there is only a small difference in the estimated costs for the best candidate solutions.

In sum, Table 8.12 indicates that it is considerably more difficult to find solutions that perform well under uncertainty with the multistage evaluation. However, the best performing stage 1 solutions have almost the same cost as in the two-stage evaluation. Based on this data, we believe that the multistage evaluation behaves differently from two-stage and that the solutions found using the latter may perform worse than anticipated if evaluated in the more realistic multistage model.

8.6.2 Solution Time

The multistage evaluation requires that the stage 2 optimization model is run at most as many times as there are ships in each scenario. For the *Medium* instance, six to eight ships are usually used. Therefore, the time it takes to evaluate a solution in stage 2 may in the worst-case increase by a factor of eight. A summary of the runtime of the framework with two-phase evaluation can be found in Table 8.13.

Table 8.13: Runtime comparison of two-stage and multistage, $\Delta T = 3$, *Medium* instance, average of 5 runs, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Evaluation	Stage 1 time [s]	Stage 2 time [s]	Solutions tested
Two-stage	2 107	5 315	191
Multistage	2 932	9 142	254

Although the runtime of the framework notably increases with the multistage rerouting, the increase is not as dramatic as one might expect. Because of the increased difficulty of finding good solutions, around 30% more solutions need to be tested before the framework terminates. This increase both the stage 1 and stage 2 runtime. The stage 2 solution time should be further increased by the fact that the multistage model needs to be run multiple time per scenario. However, the increase in stage 2 runtime is only 70%. This relatively modest increase can be explained by a favourable combination of attributes.

Firstly, the stage 2 problem becomes easier to solve with each move forward in time because more variables are locked. This effect is quite strong and the solution time approaches zero when two-three ships are looked for the *Medium* instance. Secondly, once an infeasible solution is found, the evaluation of the scenario is complete regardless of how many ships have passed the transit point. Thirdly, we do not need to solve the model as many times as there are ships. With each move forward in time, the model is only solved if any new information has been released since the previous iteration.

8.7 Analysis of Complete Framework

In this section, we want to analyse a number of different aspects of the complete framework with multistage information structure. First, we revisit the choice of ΔT in subsection 8.7.1. Then. we examine the value of flexibility in the stage 2 rerouting model in subsection 8.7.2. Third, the complete framework is tested on the *Large* instance in subsection 8.7.3. Fourth, an evaluation of the cost and robustness of the best solutions produced by the framework is presented in subsection 8.7.4. Finally, in subsection 8.7.5 we discuss the characteristics of stage 1 solutions that perform well under uncertainty.

8.7.1 Choice of ΔT

Because the behaviour of the framework changes when using multistage information structure in the evaluation, we revisit the analysis from 8.2.1 regarding the choice of ΔT. As ΔT approaches 0, the number of different solutions that can be generated approaches infinity. Thus, choosing ΔT too large could lead to good solutions not being considered. However, choosing ΔT too small could lead to too many similar solutions being evaluated. Table 8.14 presents the average results of 5 runs of the complete framework with two different values of ΔT.

Table 8.14: Multistage evaluation, *Medium* instance, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Δ T	**Solutions tested**	**Runtime framework [s]**	**Cost of best solution [NOK]**
3	254	12 074	27 905 020
1.5	978	31 953	27 931 276

Similarly to our analysis in subsection 8.2.1, the number of solutions that need to be tested before the frameworks converges more than doubles when ΔT is halved. However, the average best solution does not improve. The risk of setting ΔT too large is to miss good solutions in the candidate generation phase. As the best solutions are equally good and the solution time is significantly lower, we conclude that $\Delta T = 3$ is preferable to $\Delta T = 1.5$ in our framework.

8.7.2 Re-Routing Flexibility in Stage 2

As mentioned in subsection 8.4.1, the flexibility of the framework to re-route ships in stage 2 is controlled with the parameter ΔVisits. If ΔVisits $= 1$, each port can be visited at most once more and at least once less than in the planned schedule. Therefore, when ΔVisits $= 0$ the system's flexibility is limited to only changing the amount unloaded and the timing and sequence of visits in unloading ports. In order to analyse the impact of flexibility, the framework has been run 5 times for each value of ΔVisits. Table 8.15 presents a summary of the performance of the framework with multistage decision making. Note that column 4 denotes the percentage of potential solutions that experience one or more stock-outs in the rough evaluation, not the average percentage of stock-outs.

Table 8.15: Impact of flexibility on framework performance, $\Delta T = 3$, *Medium* instance, average of 5 runs, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

ΔVisits	**Solutions tested**	**Stage 2 time per stage 1 sol. [s]**	**Stage 1 sol. with stock-out**
0	532	6	99 %
1	227	26	86 %
2	254	36	81 %
3	248	93	87 %

The first thing to note from Table 8.15 is that it becomes extremely difficult to find solutions with no stock-outs in the challenging conditions when ΔVisits $= 0$.

However, increasing the flexibility past ΔVisits $=1$ has little impact on both the number of solutions tested before phase 1 terminates and the percentage of potential solutions with one or more stock-outs. Secondly, the runtime of the stage 2 model depends heavily on the choice of ΔVisits, as discussed in 8.4.1.

Figure 8.4 demonstrates the value of re-routing flexibility in stage 2. The 3 best candidate solutions from the thorough evaluation for 5 separate runs of the framework, a set of 15 for each value of ΔVisits, are presented as a boxplot. The estimated cost includes the penalty cost for stock-outs. As the flexibility increases, the average cost of the best solution decreases. Further, the variation in the cost of the best solutions decreases, making the performance more predictable. When increasing ΔVisits from 0 to 2, the average cost of the best solutions decreases by 1.2%. However, the effect seems to stop for further increases beyond ΔVisits$=2$.

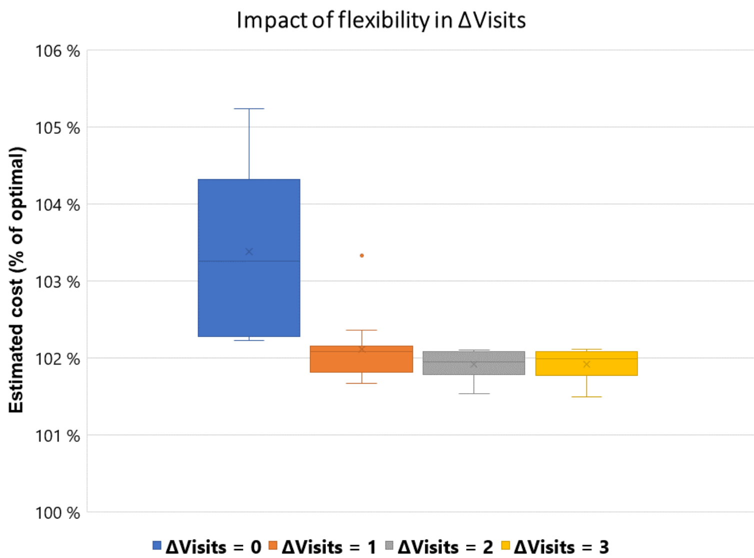

Figure 8.4: Impact of flexibility on the cost of the 3 best solutions, $\Delta T=3$, *Medium* instance, 5 runs, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

In sum, there is a clear value in allowing some flexibility to re-route ships. It becomes easier to find good schedules and the best schedules become cheaper. However, increasing ΔVisits past 2, increases the runtime significantly without improving the solutions for the *Medium* instance. We therefore argue that ΔVisits $=2$ is an appropriate choice for our framework.

8.7.3 Testing the Framework on the *Large* Instance

After the improvements made in the runtime of the framework, it may be feasible to solve larger problem instances. The *Large* instance has a time horizon of 60 days and a total of 24 ships compared to 45 days and 18 ships for the *Medium* instance. Therefore, the solution time of the optimization models is considerably longer, around 15 times the solution time of the *Medium* instance. See Table 8.8 for further details. Furthermore, when the instance size increases, the number of possible stage 1 solutions also increases.

Due to these factors, the framework does not always converge within 24 hours. Table 8.16 presents a summary of results for 5 runs of the framework with two-stage information structure. The maximum time that the framework can spend generating potential solutions and evaluating them on challenging conditions is increased from 8 to 24 hours.

Table 8.16: Framework performance, two-stage evaluation, *Large* instance, $\Delta T = 3$, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Runtime phase 1	Framework converged [%]	Cost of best solution [NOK]
8 hours	20 %	38 378 168
24 hours	60 %	38 369 770

With 24 hours to determine a set of good candidate solutions, the framework terminates with a re-chosen solution in 60% of the runs. This number is only 20% for an 8 hour runtime. However, the best solutions found in the thorough evaluation have close to identical costs. This result indicates that 8 hours is enough time to find a good set of candidate solutions for the thorough evaluation when using the two-stage information structure in stage 2.

As discussed in section 8.6, the average solution performs much worse when evaluated with multistage information structure. Therefore, it is more difficult to find good solutions and more stage 1 solutions need to be tested before the framework terminates from the candidate generation phase. The performance of the framework with multistage information structure is presented in Table 8.17. Time limits of 8 hours and 24 hours are used for the candidate generating phase. The realistic evaluation takes a further 4 hours, bringing the total runtime of the framework up to 12 and 28 hours, respectively.

Table 8.17: Framework performance, *Large* instance, $\Delta T = 3$, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

	Two-stage 8h	Multistage 8h	Multistage 24h
Solutions tested	97	154	368
Solutions with stock-out	52	146	353
Avg. simulations per sol.	8.4	4.7	4.6
Best solution [NOK]	38 378 168	38 637 907	38 668 937

The first thing to note from Table 8.17 is that more candidate solutions are tested with the multistage re-routing model than with the two-stage re-routing model. This is counter intuitive because the multistage model takes more time to solve than the two-stage model. However, the cut-off limit imposed in subsection 8.4.3 reduces the average number of simulations needed to evaluate a solution to 4.7 for the multistage model. The reduced number of simulations more than offsets the increased time per simulation and explains the results.

Secondly, it is much more difficult to find good candidate solutions with the multistage model. On average, the multistage model with 8 hours of runtime only finds 8 stage 1 solutions with zero stock-outs in the rough evaluation. For the two-stage model, this number is 45. These results indicates that the multistage model needs more time in the candidate generation phase to perform optimally. With 24 hours of runtime, the average number of solutions with zero stock-outs increases to 15.

On the other hand, the best candidate solution from the realistic evaluation performs similarly for the two models. In all 5 runs used in this test, the multistage model with 8 hours of runtime found solutions that had zero stock-outs in both the 10 challenging scenarios and in the 50 random scenarios. These results indicate that 8 hours may indeed be enough time to find at least one good solution. Furthermore, the cost of the best candidate solution does not improve when increasing the runtime to 24 hours. Based on these results, we conclude that 8 hours of runtime in the candidate generation phase is sufficient to produce good solutions for the *Large* instance. This leads to a total runtime for the framework of 12 hours for the *Large* instance.

8.7.4 Evaluation of Best Candidate Solutions

As a final evaluation of the framework, we want to analyse the performance of the best candidate solutions on an independent set of random scenarios. The objective is to determine whether the solutions chosen with the multistage extension perform notably better than the best solutions chosen by the two-stage model. To be clear, the best candidate solution refers to the stage 1 solution that performs the best in the thorough evaluation.

In order to do this, the 4 best performing candidate solutions from 5 runs of the two-stage and multistage model are collected for the *Medium* instance. The 20 solutions with the best planned deterministic cost are also included for comparison. Thus, we have a set of the 20 best solutions found using deterministic planning, two-stage information structure and multistage information structure. A similar exercise is completed for the *Large* instance where the 10 best solutions are gathered.

The results from evaluating these solutions on 50 new scenarios is presented in Figure 8.5 and 8.6. Because we believe that multistage evaluation is the most realistic, the multistage model is used to evaluate all solutions regardless of the method by which they were chosen.

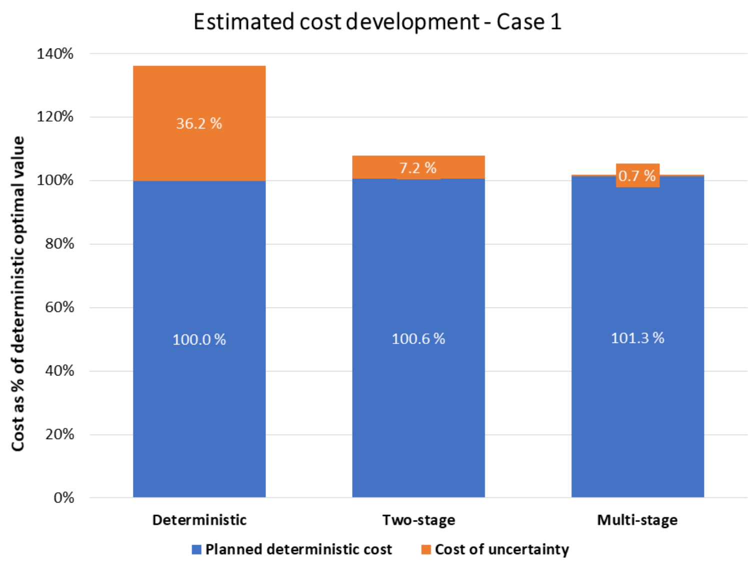

Figure 8.5: Independent evaluation of optimal solutions, *Medium* instance, $\Delta T = 3$, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Figure 8.5 paints a clear picture. The planned schedules made by the deterministic model perform poorly under uncertainty. The schedules made to perform well when all uncertainty is revealed at once performs better but the cost of uncertainty is still substantial. Finally, the multistage solutions have a slightly higher deterministic cost but performs very well under uncertainty. In fact, the multistage solutions only incur an additional cost of 0.7% due to re-routing and stock-outs compared to the plan.

The results for the *Large* instance are similar to the *Medium* instance. For a slight increase in the planned cost, the expected realized cost after uncertainty is reduced substantially with the multistage model.

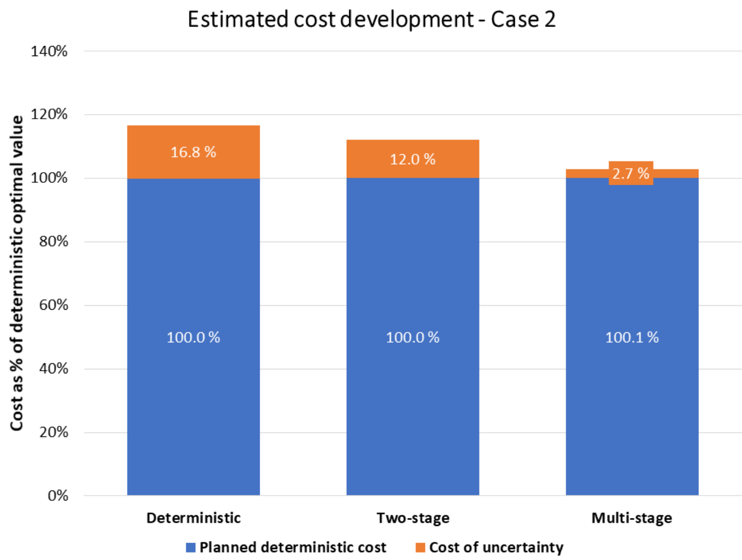

Figure 8.6: Independent evaluation of optimal solutions, *Large* instance, $\Delta T = 3$, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

The cost of stock-outs is high in our instances and infeasible solutions in stage 2 have a large influence on the total cost of uncertainty. Therefore, a further breakdown of the costs for the *Medium* instance are provided in Table 8.18. In the second column, the percentage of infeasible scenarios in the independent evaluation is presented. Columns 3 - 5 show the planned and realized costs in the scenarios that have no stock-outs.

Table 8.18: Framework performance, *Medium* instance, $\Delta T = 3$, 10 scenarios in rough evaluation, 50 scenarios in thorough evaluation

Model	% infeasible	Planned costs	Realized costs when feasible	Re-routing costs
Deterministic	46.2 %	100.0 %	104.4 %	4.4 %
Two-stage	8.3 %	100.6 %	101.7 %	1.2 %
Multistage	0.2 %	101.3 %	101.7 %	0.5 %

Table 8.18 has several interesting findings. Firstly, the realized costs in the feasible scenarios are the same for the solutions generated by two-stage and multistage re-routing. Therefore, the difference in costs in Figure 8.5 is a result of the penalty cost for stock-outs. Secondly, the multistage plans have a lower re-routing cost in the feasible scenarios than the other planned schedules. This indicates that the best stage 1 solution chosen by the multistage model results in fewer, or at least less costly, deviations from the planned schedule.

It should be mentioned that the test is skewed in favour of the multistage solutions because they have been chosen based on a multistage evaluation. Still, we believe it is interesting that the solutions chosen based on a multistage re-routing model result in significantly fewer stock-outs than the solutions chosen based on two-stage re-routing. Furthermore, this increased robustness comes at no extra cost in the scenarios that do not result in stock-outs. Finally, the cost of the best candidate solutions based on multistage re-routing does not increase in the independent evaluation compared to the thorough evaluation, indicating that the estimated cost of uncertainty is reliable.

8.7.5 Characteristics of Good Schedules

It is also interesting to compare the structure of the schedules that perform well under uncertainty to those that performs poorly. In order to do this, we have gathered the 10 best solutions under uncertainty and the 10 best deterministic solutions. All solutions considered use the same number and size of ships, and plan for the same number of departures. This is an interesting finding in itself and indicates that the best schedules under uncertainty are not that structurally different from the best deterministic schedules.

Solutions with different visit sequence in loading ports are indeed unique, but we are mostly interested in the supply to the unloading ports. Therefore, Figure 8.7 shows a box plot of the arrival time of the ships at the transit point. Note that some of these solutions may be identical because they come from different runs of the framework.

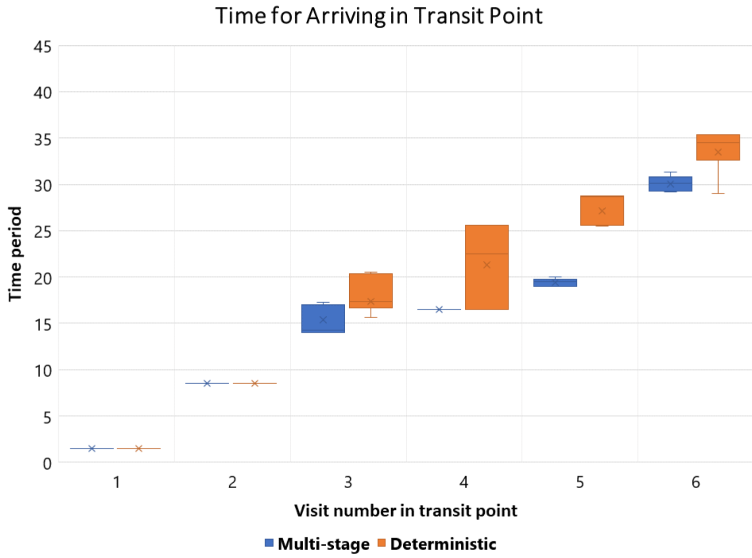

Figure 8.7: Visit time in transit point, *Medium* instance

Figure 8.7 has several interesting findings. Firstly, the good solutions have little variation in the arrival time in the transit point. Although the decisions about which ships that visit which loading ports may differ, the arrival time to the transit point varies little. Secondly, the solutions performing well plan to arrive earlier in the transit point than the poor solutions. This introduces some slack to the schedule and makes it easier accommodate delays. This result also indicates that forcing the ships to arrive earlier than necessary at the unloading ports is a good starting point for any heuristic approach to finding good schedules.

Another interesting finding relates to the number of split deliveries. The deterministic schedule planned for 1.2 split loads on average. However, in the scenarios that where not infeasible, the realized number of split deliveries was 3.1 on average. For the multistage plans, the planned number of split deliveries was 2.0, while the realized was 1.9. This result supports the findings from the previous subsection regarding re-routing costs. Incorporating some slack into the schedules leads to fewer costly split deliveries as well as less stock-outs. As noted in subsection 2.3.2, Hydro's planning division often finds it necessary to split deliveries more than planned in order to avoid stock-outs. This behaviour can also be seen in our framework for the schedules made with deterministic planning.

 96

Chapter 9

Concluding Remarks and Future Research

In this thesis, we develop a combined optimization-simulation framework to solve a maritime inventory routing problem affected by uncertainty in port admission times and sailing times. The framework iterates between generating potential solutions using an optimization model in stage 1 and evaluating their performance under uncertainty in stage 2. To capture the flexibility that Hydro has to re-route ships and change unloading quantities, the evaluation in stage 2 is done by a re-routing model that changes certain decisions to adapt the schedule after the uncertainty is revealed. The framework terminates with an estimated optimal solution when the estimated cost of a stage 1 solution under uncertainty is better than the deterministic cost of any other stage 1 solutions.

Although the original framework converges towards an estimated optimal solution, the runtime for realistic problem instances is long. In order to reduce the runtime of the framework, we consider several possible improvements. Extracting multiple solutions from each stage 1 run and evaluating them in parallel in stage 2 proves the most successful. Further, we reduce the runtime by introducing a cut-off limit to the maximum number of infeasible solutions accepted in an evaluation. The total reduction in the framework's runtime is greater than 80%.

A two-phase evaluation approach is developed to spend less time evaluating poor solutions and more time getting reliable estimates of the performance of the best solutions, called candidate solutions. This is an heuristic approach with no guarantee of finding the optimal solution. However, the results from the computational study clearly indicate that the extended framework finds more robust solutions with a lower average cost than the original framework within the same runtime.

In order to better imitate the real-world information structure, we further implement a multistage extension to the re-routing model. This model aims to capture the challenges of planning under incomplete information by not revealing all the uncertain parameters at the same time. Our results show that it is considerably more difficult to find solutions that perform well under uncertainty with the multistage re-routing. However, cost of the best solutions remains largely unchanged.

The complete framework with multistage re-routing solves a problem instance with 18 ships and a planning horizon of 45 days within 4 hours. Furthermore, our tests indicate that the planned schedules produced by our framework performs significantly better under uncertainty than both the optimal deterministic schedule and the best schedules produced with the two-stage re-routing model. For a larger instance with 24 ships and a planning horizon of 60 days, the framework does not converge. However, with a runtime of 12 hours, the framework finds stage 1 solutions with an estimated cost under uncertainty 2.8% higher than the optimal deterministic cost. This represents an estimated cost reduction under uncertainty of 14% compared to deterministic planning and 9% compared to the plans made with the two-stage information structure.

A further analysis of the performance of the solutions reveal that most of this cost reduction is a result of a reduction in stock-outs. In our case, the cost of stock-outs is high, and hence creating robust stage 1 solutions is a key priority. Furthermore, the increased robustness comes at no extra cost in the scenarios that do not result in stock-outs. The plans made with the multistage model result in fewer split deliveries than the deterministic plan and lower re-routing costs than the plans made without the multistage extension. In sum, we conclude that the framework succeeds in creating plans that have a low risk of stock-outs and costly re-routing for a marginal increase in the planned costs. Moreover, the added complexity of the multistage extension improves the performance of the schedules under realistic conditions.

Future Research

Although the iterative feedback framework performs quite well, several topics can still be investigated in future research. Firstly, other sources of uncertainty can be included in the framework quite easily by altering the random parameters drawn in each scenario.

To improve the framework, future research should address the framework's main challenge; runtime. To reduce the runtime, we have three suggestions. Firstly, parallel processing may be used more efficiently in the stage 2 evaluation, such as evaluating the different scenarios in parallel for a stage 1 solution, instead of evaluating the different solutions in parallel. Secondly, to reduce number of scenarios needed in stage 2, it could be interesting to explore an approach with a sample average approximation method. Thirdly, to decrease the number of solutions needed before convergence, future research could address how to generate potential solutions in stage 1 that are expected to perform better under uncertainty.

Furthermore, longer time horizons could be considered. The deterministic problem increases very quickly in size as the time horizon increases. Implementing the iterative framework outlined in this thesis may be challenging for such a large problem. Therefore, methods that incorporate slack into the planned schedule while only solving the model once is an interesting topic for future research. Our analysis of the best schedules in subsection 8.7.5 indicate that forcing the ships to arrive earlier than necessary is a good starting point.

References

Acar, Y., Kadipasaoglu, S. N. and Day, J. M. (2009), 'Incorporating uncertainty in optimal decision making: Integrating mixed integer programming and simulation to solve combinatorial problems', *Computers & Industrial Engineering, 56 (1)* pp. 106–112.

Agra, A., Andersson, H., Chrisitansen, M. and Wolsey, L. (2013a), 'A maritime inventory routing problem: Discrete time formulations and valid inequalities', *Networks, Vol. 62(4)* pp. 297–314.

Agra, A., Christansen, M., Delgado, A. and Hvattum, L. M. (2015), 'A maritime inventory routing problem with stochastic sailing and port times', *Computers & Operations Research 61* pp. 18–30.

Agra, A., Christansen, M., Delgado, A. and Simonetti, L. (2014), 'Hybrid heuristics for a maritime short sea inventory routing problem', *European Journal of Operational Research 236* pp. 924–935.

Agra, A., Christansen, M., Hvattum, L. M. and Rodrigues, F. (2018), 'Robust optimization for a maritime inventory routing problem', *Transportation Science 52(3)* pp. 509–525.

Agra, A., Christiansen, M. and Delgado, A. (2013b), 'Mixed integer formulations for a short sea fuel oil distribution problem', *Transportation Science, 47 (1)* pp. 108–124.

Agra, A., Christiansen, M. and Delgado, A. (2016), 'Discrete time and continuous time formulations for a short sea inventory routing problem', *Optimization & Engineering, 18 (1)* pp. 269–297.

Agra, A., Christiansen, M., Figueiredo, R., Hvattum, L. M., Poss, M. and Requejo, C. (2012), 'Layered formulation for the robust vehicle routing problem with time windows', *Lecture Notes in Computer Science, 7422* pp. 249–260.

Agra, A., Christiansen, M., Figueiredo, R., Hvattum, L. M., Poss, M. and Requejo, C. (2013c), 'The robust vehicle routing problem with time windows', *Computers & Operations Research, 40* pp. 856–866.

Al-Khayyal, F. and Hwang, S.-J. (2007), 'Inventory constrained maritime routing and scheduling for multicommodity liquid bulk, part i: Applications and model', *European Journal of Operational Research, 176 (1)* pp. 106–130.

American Chemical Society (2001), 'Hall process: Production and commercialization of aluminum', Retrieved August 28, 2018, from `http://www.acs.org/content/acs/en/education/whatischemistry/landmarks/aluminumprocess.html`.

Andersson, H., Hoff, A., Christiansen, M., Hasle, H. and Løkketangen, A. (2010), 'Industrial aspects and literature survey: Combined inventory management and routing', *Computers and Operations Research* **37**, 1515–1536.

Anthony, R. N. (1965), *Planning and Control Systems: a Framework for Analysis*, Studies in management control, Harvard University, Graduate School of Business Administration.

Ben-Tal, A., Goryashko, A., Guslitzer, E. and Nemirovski, A. (2004), 'Adjustable robust solutions of uncertain linear programs', *Mathematical Programming, 99 (2)* pp. 351–376.

Bertsimas, D. and Sim, M. (2004), 'The price of robustness', *Operations Research, 52 (1)* pp. 35–53.

Birge, J. R. and Louveaux, F. (2011), 'Introduction to stochastic programming'.

Bitran, G. R. and Tirupati, D. (1993), Hierarchical production planning, *in* S. C. Graves, A. Rinnoy Kan and P. H. Zipkin, eds, 'Logistics of Production and Inventory', 1 edn, Vol. 4 of *Handbooks in Operations Research and Management Science*, Elsevier B.V, chapter 10, pp. 523–568.

Christiansen, M. (1999), 'Decomposition of a combined inventory and time constrained ship routing problem', *Transportation Science, 33 (1)* pp. 3–16.

Christiansen, M. and Fagerholt, K. (2002), 'Robust ship scheduling with multiple time windows', *Naval Research Logistics, 4* pp. 611–625.

Christiansen, M. and Fagerholt, K. (2009), Maritime inventory routing problems, *in* C. A. Floudas and P. M. Pardalos, eds, 'Encyclopedia of Optimization', 2 edn, Springer Verlag, pp. 1947–1955.

Christiansen, M. and Fagerholt, K. (2014), Ship routing and scheduling in industrial and tramp shipping, *in* P. Toth and D. Vigo, eds, 'Vehicle Routing: Problems, Methods, and Applications', 2 edn, Springer US, pp. 381–408.

Christiansen, M., Fagerholt, K., Nygreen, B. and Ronen, D. (2007), Maritime transportation, *in* C. Barnhart and G. Laporte, eds, 'Transportation', 1 edn, Vol. 14 of *Handbooks in Operations Research and Management Science*, Elsevier B.V., chapter 4, pp. 189–284.

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

Christiansen, M. and Nygreen, B. (2005), Robust inventory ship routing by column generation, *in* G. Desaulniers, J. Desrosiers and M. M. Solomon, eds, 'Column Generation', Springer, chapter 9, pp. 197–224.

Clarksons Platou (2018), 'Research', Retrieved November 21, 2018, from http://www.platou.com/Default.aspx.

De Angelis, V., Felici, G. and Impelluso, P. (2003), 'Integrating simulation and optimization in health care centre management', *European Journal of Operational Research, 150* pp. 101–114.

Diz, G. S. d. S., Hamacher, S. and Oliveira, F. (2018), 'A robust optimization model for the maritime inventory routing problem'.

Dong, Y., Maravelias, C. T. and Jerome, N. F. (2018), 'Reoptimization framework and policy analysis for maritime inventory routing under uncertainty', *Optimization and Engineering* pp. 1–40.

Doskocz, D. (2012), 'Profitability of reduction of speed and fuel consumption for sea going bulk carriers', *Folia Oeconomica Stetinensia* pp. 132–139.

Figueira, G. and Almada-Lobo, B. (2014), 'Hybrid simulation–optimization methods: A taxonomy and discussion', *Simulation Modelling Practice and Theory, 46* pp. 118–134.

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

Fu, M. C. (2002), 'Optimization for simulation: Theory vs. practice', *Journal on Computing* pp. 192–215.

104

Groover, M. P. (2013), *Principle of Modern Manufacturing*, 5 edn, John Wiley & Sons, Inc.

Halvorsen-Weare, E. E., Fagerholt, K. and Rönnqvist, M. (2016), 'Vessel routing and scheduling under uncertainty in the liquefied natural gas business', *Computers & Industrial Engineering, 64 (1)* pp. 290–301.

Hax, A. C. and Meal, H. C. (1973), Hierarchical integration of production planning and scheduling, Working papers 656-73., Massachusetts Institute of Technology (MIT), Sloan School of Management.
URL: *https://EconPapers.repec.org/RePEc:mit:sloanp:1868*

Hemmati, A., Hvattuma, L. M., Christiansen, M. and Laportec, G. (2016), 'An iterative two-phase hybrid matheuristic for a multi-product short sea inventory-routing problem', *European journal of Operational Research, 252 (3)* pp. 775–788.

Holm, M. B. and Medbøen, C. A. B. (2017), Optimization and simulation of a maritime transportation system with transshipments at sea, Industrial economics and technology management, Norwegian University of Science and Technology.

Hydro (2012), 'Aluminium, environment and society', Retrieved August 28, 2018, from https://www.hydro.com/globalassets/1-english/about-aluminium/files/aluminium_environment-and-society.pdf.

Hydro (2017), 'Hydro 2017 market outlook', Retrieved August 28, 2018, from https://www.hydro.com/contentassets/67ae89f8f0bd44a18660ae3b48f13bef/hydro-cmd-2017-market-outlook.pdf.

Hydro (2018), 'Hydro annual report 2017', Retrieved August 28, 2018, from https://www.hydro.com/globalassets/1-english/investor-relations/annual-report/2017/downloads/annual-report-2017.pdf.

Karabakal, N., Günal, A. and Ritchie, W. (2000), 'Supply-chain analysis at volkswagen of america', *Interfaces, 30 (4)* pp. 46–55.

Ko, H. J., Ko, C. S. and Kim, T. (2006), 'A hybrid optimization/simulation approach for a distribution network design of 3pls', *Computers & Industrial Engineering, 50 (4)* pp. 440–449.

Lee, Y. H., Kim, S. H. and Moon, C. (2002), 'Production-distribution planning in supply chain using a hybrid approach', *The Management of Operations, 13 (1)* pp. 35–46.

Lide, D. R. (2005), *CRC Handbook of Chemistry and Physics, Internet Version*, 85 edn, CRC Press, Boca Raton, FL. http://www.hbcpnetbase.com.

Nikolaisen, J. B. and Vågen, S. S. (2018), Exact and heuristic approaches to a maritime inventory routing problem with inventory cost.

Nolan, R. L. and Sovereign, M. G. (1972), 'A recursive optimization and simulation approach to analysis with an application to transportation system', *Management Science, 18 (12)* .

Papageorgiou, D. J., Nemhauser, G. L., Sokol, J., Cheon, M.-S. and Keha, A. B. (2014), 'Mirplib – a library of maritime inventory routing problem instances: Survey, core model, and benchmark results', *European Journal of Operational Research 235* pp. 350–366.

Port of Bergen (2018), 'Port charges', Retrieved October 20, 2018, from https://bergenhavn.no/wp-content/uploads/2018/02/prices-2018_EN.pdf.

Rakke, J. G., Andersson, H., Christiansen, M. and Desaulniers, G. (2015), 'A new formulation based on customer delivery patterns for a maritime inventory routing problem', *Transportation Science, 49 (2)* pp. 384–401.

Rakke, J. G., Stålhane, M., Moe, C. R., Christiansen, M., Andersson, H., Fagerholt, K. and Norstad, I. (2011), 'A rolling horizon heuristic for creating a liquefied natural gas annual delivery program', *Transportation Research Part C, 19* pp. 896–911.

Rio Tinto (2018), 'Aluminium projects and operations', Retrieved August 28, 2018, from https://www.riotinto.com/aluminium-83.aspx#map/operations.

Sea-distances.org (2018), 'Sea distances', Retrieved October 20, 2018, from https://sea-distances.org/.

Ship and Bunker (2018), 'World bunker prices', Retrieved October 20, 2018, from https://shipandbunker.com/prices.

Siswanto, N., Essam, D. and Sarker, R. (2011), 'Solving the ship inventory routing and scheduling problem with undedicated compartments', *Computers & Industrial Engineering, 61 (2)* pp. 289–299.

Soyster, A. L. (1973), 'Convex programming with set-inclusive constraints and applications to inexact linear programming', *Operations Research, 21 (5)* pp. 1154–1157.

Stålhane, M. (2013), Optimization of Maritime Routing and Scheduling Problems with Complicating Inter-route Constraints, PhD thesis, Norwegian University of Science and Technology.

The International Aluminium Institute (2018), 'World aluminium statistics', Retrieved September 29, 2018, from http://www.world-aluminium.org/statistics/.

Wikipedia (2018), 'Bulk carrier', Retrieved November 21, 2018, from https://en.wikipedia.org/wiki/Bulk_carrier.

Zhang, C., Nemhauser, G., Sokol, J., Cheon, M.-S. and Keha, A. (2018), 'Flexible solutions to maritime inventory routing problems with delivery time windows', *Computers and Operations Research, 89* pp. 153–162.

Sofie S. Vågen & Jørgen B. Nikolaisen

Master's Thesis

NTNU
Kunnskap for en bedre verden

Hydro