

Carina Nygård

An Investigation of Methods and Value of Information in Routing of Priority-Based Operations

Using a Rule-Based Routing Method Tested with Discrete Event Simulation

Master's thesis in Marine Technology

Supervisor: Bjørn Egil Asbjørnslett

June 2020

MASTER THESIS IN MARINE TECHNOLOGY

SPRING 2020

For stud.techn.

Carina Nygård

An Investigation of Methods and Value of Information in Routing of Priority-Based Operations

Using a Rule-Based Routing Method tested with Discrete Event Simulation

Spring 2020

Background

In the next 30 years, the world population is expected to increase with 2 billion. A growing population leads to an increased need for energy and food. Available land areas for energy and food production are decreasing, and we are therefore forced to look towards the oceans.

The coastal zone can be used for several purposes, such as fisheries, oil and gas, aquaculture, and wind energy. The fisheries have been there for centuries and are still an important industry for food production. The oil and gas industry has been, and still are, an essential part of many countries' energy production and economy. The aquaculture industry is predicted to be crucial in the global strategy for ensuring enough food for a growing world population. At the same time, offshore wind energy is important in the upcoming energy shift. As a consequence, the pressure on coastal zones has increased, and the industries are looking at the possibility of moving the operations to more exposed locations. The fish farming company Salmar has already launched the world's first offshore fish farm, Ocean Farm 1, placed in more exposed waters off the coast of Norway. Equinor has launched Hywind Scotland, the world's first floating wind farm. Both of these pave the way for new and innovative solutions in more exposed waters, while still conducting operations in sheltered waters as well.

As industries are pursuing operations in more exposed waters, the circumstances regarding the operations change. More exposed locations represent increased possibilities and suitable metocean conditions, but they also lead to increased technical requirements for both the installation, the operations, and the vessels used. Operations in areas with higher exposure to weather, therefore, require a vessel capable of handling longer periods with tough weather conditions. Special vessels have certain operational limits, deciding when an operation can be conducted safely. Both aquaculture and wind energy companies have facilities placed in areas with various exposure. The requirements for conduction operations in these facilities may vary, both with regards to the operational limits of the vessel and planning time. The companies must, therefore, choose whether to use the same vessel for all operations or different vessels for each of the facilities with various exposure.

The use of one special vessel for facilities placed in areas with different exposure requires efficient routing of the vessel. The routing must take several factors that can affect the efficiency into

consideration, such as priority, metocean conditions, and demands at each location. More exposed locations have a higher probability of unavailability due to the weather conditions, and they must, therefore, be prioritized before the normal and sheltered locations. In short, the locations are assigned various priorities based on weather exposure and the probability of unavailability. The facilities mentioned, such as fish farms and wind turbines, also have demands that must be complied with in a routing situation. The routing of the special vessel must, therefore, consider the priority of the locations, demands at each location, and metocean conditions. This kind of problem has not, as the author knows, been investigated before. In order to gain insight into such a problem, this master thesis will look at how the routing should be made and the value of information when routing one special vessel. Value of information can be described as the sum a decision-maker would be willing to pay for information about the future before making a decision and the benefit gained by knowing the future.

Objective and focus

The routing of a vessel capable of handling operations in areas with different exposure and priority is a time consuming and large problem. In order to gain insight into this problem, this master thesis will look at various methods and the value of information when routing one vessel between facilities with various exposure.

Gaining insight into the problem should be achieved by providing a detailed problem description, locating factors essential for routing of priority-based operations, evaluating various methods for routing, developing a method for the routing of the vessel, and testing the method with the use of discrete event simulation.

Scope and main activities

The candidate should presumably cover the following main points:

- 1. Explore the problem and establish a detailed problem description.*
- 2. Provide a short overview of the current status and important development trends related to routing and scheduling of priority-based operations.*
- 3. Review relevant routing methods for this kind of problem.*
- 4. Review the necessary theory.*
- 5. Identify crucial factors for routing of priority-based operations.*
- 6. Develop a method that generates schedules for the routing of a vessel that meet customer demands.*
- 7. Develop a simulation model that test the method which generates the schedules for the vessel and implicitly assess the value of information.*
- 8. Test robustness of the routing method by including weather uncertainty in the simulation.*
- 9. Perform a case study and test the routing method. Test with various levels of uncertainty and information.*
- 10. Discuss and conclude*

Modus operandi

At NTNU, Professor Bjørn Egil Asbjørnslett will be the responsible advisor, and PhD Candidate Hans Tobias Slette will be co-supervisor.

The work shall follow the guidelines given by NTNU for the MSc Project work

Bjørn Egil Asbjørnslett
Professor/Responsible Advisor

Hans Tobias Slette
PhD Candidate/Co-supervisor



NTNU

Norwegian University of
Science and Technology

MASTER THESIS

**AN INVESTIGATION OF METHODS AND VALUE OF
INFORMATION IN ROUTING OF PRIORITY-BASED
OPERATIONS**

USING A RULE-BASED ROUTING METHOD TESTED WITH DISCRETE EVENT SIMULATION

Author:

NYGÅRD, Carina

Supervisor:

ASBJØRNSLETT, Bjørn Egil

Co-supervisor:

SLETTE, Hans Tobias

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF ENGINEERING

DEPARTMENT OF MARINE TECHNOLOGY

Preface

This master thesis represents the finalization of my Master of Science degree in Marine Technology at the Norwegian University of Science and Technology in Trondheim. The report is written during the spring of 2020 and counts for 30 credits. It should be noted that the project thesis handled a completely different topic, and the master thesis is, therefore, the result of only five months of work. A significant portion of the thesis was written while in lock-down due to Covid-19.

Since the problem considered has received limited attention, the process of defining and understanding the problem has been time-consuming. The routing method and simulation model has been developed in parallel with the investigation of the problem. The thesis touches upon several aspects focused on in my specialization program in Marine Systems Design and Logistics, such as routing of vessels, optimization, programming and simulation.

I want to thank my supervisor, Professor Bjørn Egil Asbjørnslett, who introduced me to the thesis problem and provided both motivational and professional support during the work with the master thesis. I would also like to thank my co-supervisor, Ph.D. Candidate Hans Tobias Slette, for valuable professional discussions, motivation, and help with the model.

I would also like to thank my family for always having my back and encouraging me throughout my studies at NTNU. A final thank you goes to my classmates for valuable inspiration, much-needed motivation, and loads of fun during our years at Tyholt.

Trondheim, June 15th, 2020



Carina Nygård

Abstract

Operations in the coastal zone are moving towards more exposed waters. A problem may arise when operations in exposed and sheltered waters intends to use the same vessel. Facilities, such as ocean farms, in exposed waters have a higher probability of unavailability due to weather conditions, and are therefore assigned a higher priority compared to the facilities in sheltered waters. This priority applies to all the operations that should be conducted at exposed locations. Since one vessel operates at all locations, an efficient routing is required. The objective of this thesis is to look at various methods for routing of priority-based operations and the value of information in such routing. The findings may be used in further analysis of the problem, and a decision support tool for relevant industries, such as the aquaculture or the wind energy industry.

In order to understand the problem, crucial factors for routing of priority-based operations, such as customer demands, weather conditions and the priority of each location, are assessed. In order to find a suitable routing method, various methods are reviewed before a rule-based method is chosen. The rule-based method consists of a chosen set of rules deciding where the vessel should sail. Four rule sets are generated in MATLAB, where the difference between them is the amount of information available. Each rule set build on the previous set with regards to information. Rule set 1 only knows the consumption rate and storage capacity at each facility. Rule set 2 knows the weather forecast for the next day in addition to the information given in Rule set 1. Rule set 3 knows the weather forecast for the next two weeks in addition to whats given in Rule set 1, while Rule set 4 also knows the inventory levels at each facility. To test the routing method and assess the value of information, a case study of the feed distribution at three facilities with various exposure is conducted. The rule sets are tested using a discrete event simulation with four degrees of uncertain weather conditions. The simulation model is built in Simulink and simulate one year of operation for the three locations, taking weather conditions, priority of each location and feed levels into account. Weather forecasts are simulated with the use of Markov chains and random numbers.

The simulation model calculates the total cost of one year of operation, in addition to monitoring change in inventory levels, amount of feed delivered and sea states experienced. The total cost can be divided into charter, fuel, and delay costs. The cost components are used to evaluate the performance of each rule set and the value of information. The value of information can be described as the benefit gained by including extra information in the decision process, such as adding exact weather data or updated inventory levels. The results indicate a high value of information in knowing the inventory levels, while the weather forecast gives surprisingly small benefits. The simulation also showed that a rule-based method accomplish to book visits in an acceptable fashion, but it is not believed to be optimal due to low utilization of the vessel.

Sammendrag

Operasjoner i kystsonen beveger seg mot mer eksponerte farvann. Et problem kan derfor oppstå når operasjoner i eksponerte og skjermede farvann har til hensikt å bruke det samme fartøyet. Fasiliteter, slik som for eksempel havmerder, i utsatte farvann har større sannsynlighet for utilgjengelighet på grunn av værforhold, og tildeles derfor en høyere prioritet sammenlignet med fasilitetene i skjermet farvann. Denne prioriteringen gjelder alle operasjoner som skal utføres i slike områder. Siden ett fartøy opererer innefor områder med ulik eksponering for vær, så er det nødvendig med effektiv ruteplanlegging. Målet med denne masteren er å se på ulike metoder for ruteplanlegging av prioriteringsbaserte operasjoner og verdien av informasjon i en slik ruteplanlegging. Funnene kan brukes i videre analyse av problemet, og et beslutningsstøtteverktøy for relevante næringer, som for eksempel havbruk- eller vindkraftindustrien.

For å forstå problemet undersøkes avgjørende faktorer for ruteplanlegging av prioriteringsbaserte operasjoner, slik som kundekrav, værforhold og prioritering av hver lokasjon. Ulike relevante metoder for ruteplanlegging gjennomgås før en regelbasert metode velges. Den regelbaserte metoden består av et valgt sett med regler som bestemmer hvor fartøyet skal seile. Fire regelsett genereres i MATLAB, hvor forskjellen mellom dem er mengden tilgjengelig informasjon. Hvert regelsett bygger på det forrige settet med hensyn til informasjon tilgjengelig. Regelsett 1 kjenner kun til forbruksraten og lagringskapasiteten på hvert anlegg og på skipet. Regelsett 2 kjenner til værmeldingen for neste dag i tillegg til informasjonen gitt i regelsett 1. Regelsett 3 kjenner til værvarselet for de neste to ukene i tillegg til det som er gitt i regelsett 1, mens regelsett 4 også kjenner lagernivåene på hvert anlegg. For å teste ruteplanleggingsmetoden og vurdere verdien av informasjon, blir det gjennomført en casestudie av fôrdistribusjonen ved tre anlegg med forskjellig eksponering. Regelsettene testes ved bruk av en diskret hendelsessimulering med fire nivå med usikre værforhold. Simuleringsmodellen er laget i Simulink og simulerer ett års drift for de tre stedene, og tar hensyn til værforhold, prioritering av hvert sted og fôrnivå. Værmeldinger simuleres ved hjelp av Markov-kjeder og tilfeldige tall.

Simuleringsmodellen beregner de totale kostnadene for ett års drift, i tillegg til å overvåke endring i lagernivå, mengde levert fôr og opplevde havtilstander på hver lokalitet. Den totale kostnaden kan deles inn i charter-, drivstoff- og forsinkelseskostnad. Kostnadskomponentene brukes til å evaluere ytelsen til hvert regelsett og verdien av informasjon. Verdien av informasjon kan beskrives som fordelene oppnådd ved å inkludere tilleggsinformasjon i beslutningsprosessen, for eksempel ved å vite eksakte værdata eller oppdaterte lagernivåer for hver lokalitet. Resultatene fra casestudien indikerer en høy verdi av informasjon ved å kjenne til lagernivået, mens værmeldingene gir overraskende små fordeler. Simuleringen viste også at en regelbasert metode planlegger besøk til hver lokasjon på en akseptabel måte, men den antas ikke å være optimal på grunn av lav utnyttelse av fartøyet.

Table of Contents

Preface	i
Abstract	iii
Sammendrag	v
Table of Contents	ix
List of Tables	xi
List of Figures	xiv
List of Algorithms	xv
Abbreviations	xvii
1 Introduction	1
1.1 Background and Motivation	1
1.2 State of Art	2
1.3 Objective	3
1.4 Report Structure	3
2 Problem Description	5
2.1 What is the problem?	5
2.2 Problem Definition	5
2.3 Factors influencing the Routing	8
2.4 Cost Function	8
3 Literature Review	11
3.1 Priority and Value of Information	11
3.2 Exact Methods	11
3.3 Approximate Methods	13
3.4 Rule-based Methods	14
3.5 Conclusion of Literature Review	14

4	Methodology	17
4.1	Possible Methods	17
4.1.1	Exact Method	17
4.1.2	Large Neighborhood Search (LNS)	18
4.1.3	Rule-based Method	20
4.1.4	Choice of Method	21
4.2	Rule Set Formulation	21
4.2.1	Exploring the Problem	21
4.2.2	Proposing Rule Sets	26
4.3	Exploring Uncertainty through Simulation	28
4.3.1	Simulink and SimEvents	29
4.3.2	Weather Representation	29
4.3.3	Markov Chain Simulation	30
4.4	Value of Information	30
5	Testing Rule Sets with Simulation	33
5.1	Model Setup	33
5.2	Start of Model	33
5.3	Model Input	34
5.3.1	Vessel Characteristics and Parameters	34
5.3.2	Weather Conditions	35
5.3.3	Rule Set Implementation and Plan Generation	36
5.4	Simulation Model	37
5.4.1	Entity Generation	38
5.4.2	Global Variables	38
5.4.3	Routing	39
5.4.4	Re-Routing	40
5.5	Model Output	41
5.6	Verification of Model	41
6	Case Study	43
6.1	Case Description	43
6.2	Case Details	44
6.3	Weather Data	45
6.3.1	Various Realizations	46
6.4	Results from Simulation and Evaluation of Rule Sets	48
6.4.1	Total Cost	48
6.4.2	Simulation Output	49
6.5	Evaluation of Value of Information	57

7 Discussion	59
7.1 Modelling Assumptions	59
7.2 Simulation Model	61
7.3 Simulation Results	63
8 Conclusion	65
8.1 Further Work	65
Bibliography	66
Appendix	70
Appendix A Mindmap	I
Appendix B Case A	III
Appendix C Case B	V
Appendix D Case C	IX
Appendix E Rule sets	XI
E.1 <i>Ruleset111</i>	XI
E.2 <i>Ruleset112</i>	XII
E.3 <i>Ruleset113</i>	XIV
E.4 <i>Ruleset114</i>	XV
Appendix F Simulation Model	XIX
F.1 Routing Model up close	XXI
F.2 MATLAB function	XXII
F.3 Global Variables	XXII
Appendix G MATLAB Scripts	XXV
G.1 <i>run_simulation</i>	XXV
G.2 <i>WeatherMatrixOneYear</i>	XXXIII
G.3 <i>availabilityWeatherMatrix</i>	XXXIX
G.4 <i>MATLAB Function2NewRuleSets</i>	XXXIX
Appendix H Simulation Output	XLVII

List of Tables

2.1	Classification of location type and level of priority.	6
2.2	Summary of system parameters.	7
4.1	Description of objective function and constraints relevant for a mathematical formulation of the thesis problem.	18
4.2	Comparison of possible methods.	21
4.3	Conditions for Case A scenarios.	23
4.4	Conditions for Case B scenarios.	24
4.5	Description of information available in each rule set	26
4.6	Example of a Markov chain transition matrix.	30
5.1	Ship attributes.	34
5.2	Cost and inventory parameters.	34
5.3	Format of the plan generated.	36
5.4	Vessel attributes with initial value.	38
5.5	Description of all global variables in the model.	39
6.1	Description of cases.	44
6.2	Left: Overview of distances between the different locations in nautical miles. Right: Ship characteristics.	44
6.3	Parameter values in case study.	45
6.4	Sea state ranges and their associated probabilities.	46
6.5	Transition matrix generated based on historical data.	46
6.6	Summary of the total cost in NOK for each simulation. The four values in each cell are from simulations with Seed 1, 2 and 3, and their average value.	48
6.7	Summary of simulation results for Rule set 1, Case 1,2, 3 and 4 with Seed 2.	49
6.8	Summary of simulation results for Rule set 2, Case 1,2, 3 and 4 with Seed 2.	51
6.9	Summary of simulation results for Rule set 3, Case 1,2, 3 and 4 with Seed 2.	53
6.10	Summary of simulation results for Rule set 4, Case 1,2, 3 and 4 with Seed 2.	55

List of Figures

2.1	Illustration of the problem.	6
4.1	Illustration of the principle of destroy and repair, with inspiration from (Pisinger and Ropke, 2010). The left figure shows a VRP solution before the destroy operation and the middle figure after 3 customers are removed. The right figure show the solution after the repair operation.	19
4.2	Factors influencing the choice of route.	22
4.3	Illustration of simple model used in manual case analysis.	22
4.4	Example of Test A procedure.	23
4.5	Example of Test B procedure.	24
4.6	Markov chain transition diagram.	30
5.1	Flow chart for the overall model.	33
5.2	Illustraion of weather matrix.	35
5.3	Snapshot from Simulink model illustrating the four different places to go; exposed location, normal location, sheltered location and port.	37
5.4	Illustration of the blocks used to define a global variable.	38
5.5	Port block in simulation model.	40
5.6	Illustration of the MATLAB function used for the rerouting and generation of new plan.	41
6.1	Map over aquaculture facilities in case study. Map from Kystverket (2020).	43
6.2	Probability distribution of the H_g values included in the data set.	46
6.3	Sea states for exposed in column one and normal in column two. Row one is for Seed 1, row two for Seed 2 and row three for Seed 3.	47
6.4	Total cost and delay cost for each of the locations for Rule set 1, Case 2, Seed 2.	50
6.5	Total cost and delay cost for each of the locations for Rule set 2, Case 2, Seed 2.	52
6.6	Delay cost for each of the locations for Rule set 4, Case 2.	54
6.7	Total cost and delay cost for each of the locations for Rule set 4, Case 2, Seed 2.	56
6.9	Histogram of the amount of feed transported each day in operation for (a) Rule set 1, (b) Rule set 2, (c) Rule set 3 and (d) Rule set 4.	57
B.1	Case A.	III

C.1	Case B	VI
C.2	Case B log.	VII
C.3	Continue, Case B log.	VIII
D.1	Case C details.	IX
D.2	Case C log.	X
F.1	Snapshot of the complete model in Simulink.	XX
F.2	Simulation model, snapshot from Simulink.	XXI
F.3	MATLAB function used for rerouting.	XXII
F.4	Global variables for cost components.	XXII
F.5	Global variables for number of visits.	XXIII
F.6	Time generator with associated global variable.	XXIII
F.7	Global variables for inventory levels.	XXIII
F.8	Plan overview, global variable and display.	XXIV

List of Algorithms

1	Pseudocode for the adaptive large neighborhood search method.	19
2	Pseudocode for the rule-based routing algorithm.	20
3	Pseudocode Rule set 1.	27
4	Pseudocode Rule set 2.	27
5	Pseudocode Rule set 3.	28
6	Pseudocode Rule set 4.	28

Abbreviations

ALNS	=	Adaptive Large Neighborhood Search
IRP	=	Inventory Routing Problem
LNS	=	Large Neighborhood Search
MPP	=	Maritime Planning Problems
OR	=	Operation Research
PDPTW	=	Pickup and Delivery Problem with Time Windows
PSVPP	=	Periodic Supply Vessel Planning Problem
PVRP	=	Periodic Vehicle Routing Problem
RS	=	Rule set
RSPMFOWF	=	Routing and Scheduling Problem of a Maintenance Fleet for Offshore Wind Farms
SSS	=	Short Sea Shipping
SVPP	=	Supply Vessel Planning Problem
SVRPPD	=	Single Vessel Routing Problem with Pickups and Deliveries
SVWRP	=	Supply Vessel Weather Routing Problem
TSP	=	Traveling Salesman Problem
VMI	=	Vendro Managed Inventory
VoI	=	Value of Information
VRP	=	Vehicle Routing Problem
VRPPDTW	=	Vehicle Routing Problem with Pickup and Delievery and Time Windows

1 | Introduction

In the following section, the background and motivation for the thesis will be described before the state of the art and objective is presented. Lastly, the structure of the thesis is described.

1.1 Background and Motivation

The society is reacting to a rapidly increasing world population. The world population is estimated to increase by 2 billion in the next 30 years, reaching 9.7 billion in 2050, (UN, 2019). As the population grows, the need for energy and food increases. The remaining land areas for food and energy production are decreasing, and the society is forced to look elsewhere.

The coastal zone can be used for several purposes, such as fisheries, oil and gas, aquaculture, and wind energy. The fisheries have been there for centuries and are still an important industry for food production. The oil and gas industry has been, and still is, an essential part of many countries' energy production and economy. The fish farming industry is predicted to be crucial in the world wide strategy of ensuring enough food for a growing world population. The offshore wind energy will be essential in the upcoming energy shift, and the number of turbines is likely to grow. As a result, the pressure on the coastal zones has increased, and the industries are consequently looking for new areas to utilize. As a consequence, they are looking at the possibility of moving parts of the facilities further offshore in more exposed waters. The fish farming company Salmar has already launched the world's first offshore fish farm, Ocean Farm 1, placed in more exposed waters off the coast of Norway. This farm was the first ocean farming cage in operation, built with a development licence given by the Norwegian authorities. Ocean Farm 1 was the start of a new era and has inspired several other developments within offshore fish farms. In the wind energy sector, companies are partly shifting their focus from bottom fixed wind turbines to floating. The energy company Equinor launched Hywind Scotland in 2017, the world's first floating wind farm, and will launch a new farm, Hywind Tampen, in 2022. Both of these companies pave the way for new and innovative solutions in more exposed waters, while still conducting operations in sheltered waters as well.

As industries are pursuing operations in more exposed waters, the circumstances regarding the operations change. More exposed locations represent increased possibilities and suitable metocean conditions, but they also lead to increased technical requirements for both the installation, the operations, and the vessels used. Operations in areas with higher exposure to weather, therefore, require a vessel capable of handling longer periods with tough weather conditions. Special vessels have certain operational limits,

deciding when an operation can be conducted safely. Both aquaculture and wind energy companies have facilities placed in areas with various exposure. The requirements for conduction operations in these facilities may vary, both with regards to the operational limits of the vessel and planning time.

Facilities within the aquaculture industry are spread out in areas with different exposure, especially in Norway. Some farms are located in sheltered areas inshore with no exposure, while others are located in areas with some exposure. The same vessel is often used when, for example, delivering feed to each facility. Now that fish farms can be placed in exposed areas as well, there might be a question of whether to use two different vessels or design and build one vessel capable of handling operations at all types of locations. The potential savings made by practicing one ship for all areas can be interesting.

Areas with a higher exposure are more prone to bad weather than sheltered areas and, therefore, have a higher probability of unavailability. The difference in weather conditions in the various areas necessitate a priority for each of the locations. An exposed location should be prioritized before the more sheltered when available, since the likelihood of unavailability is higher. These priorities must be taken into consideration during the routing of the vessel.

The difference in weather exposure can pose a challenge for the routing of the supply vessel if the same vessel is used to supply locations with different exposures. Route planning of vessels used in marine operations have received little attention through the years, but are becoming more important now that the environmental focus is increasing. Earlier methods for routing of vessels with regards to weather are based on experience or manual decisions. There are therefore large opportunities for improvement.

Route planning where the weather is considered is called "weather routing". Weather routing has been in focus since the early sixties, but little research on the effects of weather routing has been conducted. Weather routing can save companies for substantial operational costs and can, therefore, be very useful, especially now that weather forecasts are becoming more detailed and correct.

Apart from routing based on weather, it is also common to route a vessel based on customer demand. Alongside the priority, each of the facilities in the aquaculture industry has needs that must be fulfilled. Feed must be delivered, and operations must be conducted. The same applies to the wind industry, where each wind turbine has needs that must be fulfilled with regards to maintenance. Regular visits must be made to most facilities placed in the ocean to keep the construction up to date and make sure the operation is safe.

1.2 State of Art

Routing of a vessel with regards to priority, customer demands, and weather conditions has received little attention through the years, but a higher focus on efficiency in the last few years has increased the focus. The survey by Haupt (1989) look at priority in job shop scheduling. Szlapczynski (2006) includes a priority policy in the routing of various vessels. Agra et al. (2017) considered a maritime inventory routing problem for a salmon farmer in Norway. Fagerholt et al. (2013) look at the project shipping problem for a shipping company and provide a tabu search heuristic which give significant improvements compare to manual planning. Grifoll et al. (2018) look at the potential economic benefits of using a weather ship routing system for short sea shipping. Sandvik et al. (2019) develops a model

capable of controlling operational and tactical navigation decision, based on weather and ship resistance. The mentioned articles touches upon areas relevant for routing of a vessel based on demand, weather and priority. A specific method that takes priority into consideration when routing a vessel has, to the author's knowledge, not been reviewed before. The value of information in such a model will, therefore, be investigated.

1.3 Objective

The objective of this master thesis is to gain insight into the routing of a vessel between locations with different exposure and, therefore, also priority. In order to gain insight, various methods and the value of information when routing one vessel between facilities with various exposure will be investigated. This should be achieved by exploring the problem definition, locating important routing factors, developing a method for the routing of the vessel, and testing the method with the use of discrete event simulation.

1.4 Report Structure

The remainder of this thesis is organized as follows. In Chapter 2, the problem is described before Chapter 3 reviews various methods previously used in related problems that can be used to solve the thesis problem. In Chapter 4, a selected set of methods is explored before a final method is chosen. This method is further explained, and relevant theory is presented. In Chapter 5, the model is presented and explained before Chapter 6 presents the case study covering feed distribution. Both case description and weather data used are explained before the results from each case are presented and discussed. In Chapter 7, the assumptions made while making the model and the results from the simulation are discussed before a conclusion is made, and further work is proposed in Chapter 8.

2 | Problem Description

In the following section, the problem will be defined and described before factors influencing the routing are presented. Lastly, the cost function that will be used to assess the routing method is described.

2.1 What is the problem?

When routing a vessel, there are several things to take into consideration in order for it to be cost efficient. An inefficient routing can be costly. A vessel can, for example, sail to a facility without checking the weather. When the vessel arrives, the wave heights are too big for the operation to be conducted. The vessel has then wasted time and money by travelling to a location that was unavailable. The vessel then has three choices; wait for the weather to clear up, sail back to port or sail to another location. This choice is dependent on the amount of information available. If the vessel captain had checked the weather forecast in advance, it would probably indicate that the location was unavailable and the vessel could have sailed somewhere else.

Another example of an inefficient routing is if a vessel travels to a facility to deliver fish feed, and only 500 kg of feed is delivered. Compared to the 2000 tons of storage capacity, this is a low amount and the cost of sailing with such a small amount is high. If the fish farmer had updated the captain with the feed levels, he could have made a different call.

And what if two locations are available, they have different levels of demand and different priorities. Should the priority always decide where to go, or can the demand beat the priority in certain scenarios?

These are examples of inefficient routing and problems that may arise which can be avoided with good planning and information about the customer demands, weather and the priority of each location. The problem will be further defined in the following sections.

2.2 Problem Definition

The problem explored in this thesis is the delivery of supplies to a set of customers in regions with various exposure to weather, taking priority into consideration. Weather in this context can be interpreted as wind, waves, and current that may exceed the operational limits of a vessel. A set of customers N has a demand D that must be satisfied. The demand can be in the form of an operation that must be completed which demands a visit to the location. In this case, the demand from each customer is in the form of supplies. The system consists of three customers, placed in different regions in the ocean, and one vessel that must visit these customers. The difference between the regions is the degree of exposure

to weather, which affects the ability to operate. Three degrees of exposure are defined; sheltered, normal, and exposed. In the sheltered location, operations may always be performed. In a normal location, the operation is limited by the weather to some degree. In the exposed location, the weather is highly limiting for the operation to be completed. The exposed location is located further from port compared to the normal and sheltered location, as illustrated in Figure 2.1. The vessel starts in port and has the ability to visit one or several locations during one voyage.

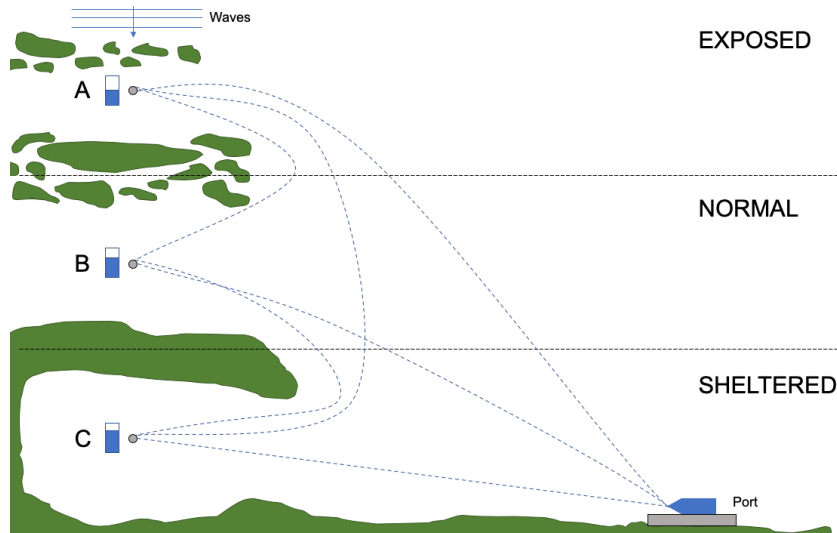


Figure 2.1: Illustration of the problem.

The difference in exposure is determined by the probability of bad weather at each location. Exposed areas have a higher probability of bad weather. Bad weather can be interpreted as weather conditions where the vessel cannot perform the required operations necessary. A location is, in this thesis, classified as exposed if the probability of bad weather on a random day during the year is above 20%. If the probability of bad weather is under 20%, the location can be classified as normal. A sheltered location is set to have a 0% chance of bad weather.

The variation in weather conditions at each location induce a difference in priorities. An exposed location is prioritized before a normal and sheltered location due to the likelihood of the location being unavailable due to weather conditions. This priority must be included in the routing decision process. A summary of the classifications can be seen in Table 2.1.

Classification	Probability of bad weather	Priority level
Exposed	> 20%	High
Normal	< 20%	Medium
Sheltered	0%	Low

Table 2.1: Classification of location type and level of priority.

Each location has an inventory storage with an inventory capacity of I_{cap} and a consumption rate r . The customers demand that the inventory level at each location does not go to zero. If the inventory level

go to zero, a monetary punishment C_{delay} is charged for each day under the limit. The model should generate a schedule that complies with the demand for each customer for a two week planning period. During this period, the schedule should be updated based on a change in weather forecast. If the location is not available due to weather, the visit should not be completed.

The problem is studied only for cases where one vessel is sufficient to serve the locations. The chosen vessel has a service speed V and a storage capacity S_{cap} . It can be used for operations at all locations, but it has an operational limit, OP_{lim} , that must be adhered to. If the wave height at the site exceeds the operational limit of the vessel, it cannot go through with the operation.

The vessel starts in port and must travel a given distance A_{ij} from port i to location j for operation. The operational hours of the vessel are from 06:00 to 18:00. Loading in port must be completed within this time period and require a total time, T_{load} , of one hour. The loading may be conducted a day in advance if the vessel is back in port before 17:00. The vessel can visit several customers if it manages to be within the time restriction.

The stakeholder in the problem is the owner of the facilities which must be served. They charters the vessel by a day rate, C^{TC} , and is interested in minimizing the chartering costs. The vessel is assumed to be used at other locations outside this problem, which support the importance of efficient operations in order for the vessel to be available for other locations as well. In addition to the time charter rate, a fuel cost, C^F , must be considered. A summary of all the mentioned parameters describing the problem can be seen in Table 2.2.

Description of parameter	Parameter	Unit
Set of customers	N	[-]
Priority of location i	P_i	[-]
Demand at location i	D_i	[ton]
Inventory capacity at location i	I_{cap}	[ton]
Lower inventory limit	I_{low}	[ton]
Inventory consumption rate	r	[ton/day]
Speed of vessel	V	[knots]
Storage capacity vessel	Q_{max}^v	[ton]
Operational limit	OP_{lim}	[m]
Charter cost	C^{TC}	[NOK]
Fuel cost	C^F	[NOK]
Delay cost at location i	C_i^D	[NOK]
Loading time in port	T_{load}	[h]
Distances	A_{ij}	[nm]

Table 2.2: Summary of system parameters.

2.3 Factors influencing the Routing

There exist various types of information that can be used in the decision process of where and when the vessel should sail. As mentioned, such factors can be the priority of the location, weather conditions at the site, inventory levels, and cost. In order for the reader to understand the complexity of the problem, some simple scenarios will be demonstrated.

A vessel routing may have several possible approaches. The problem described, with three different locations, is used as an example, similar to Figure 2.1, but with equal exposure. Let us say the vessel must decide where to go on a random day during the year. The locations are identical, but location A has a lower inventory level than B and C. Based on the inventory level, the vessel chooses to sail to A. If the weather is also considered, A may not be available due to large wave heights. Then the vessel must choose between B and C. If B and C are given different priorities, this must be taken into consideration, together with the weather conditions and the inventory levels. In addition to the mentioned factors, other factors such as the distances, delay costs, and the storage capacity, may also influence the decisions made by the captain.

There are, in other words, a considerable number of factors that must be included in the decision process when routing the vessel in this case. The factors do not only complicate the problem, but they are also believed to assist in providing better and more robust solutions. The factors should, therefore, be understood and included in the routing model.

2.4 Cost Function

There are three different costs related to this problem; the cost of chartering, fuel cost and delay cost. The sum of these will constitute the total cost of the routing. A part of the objective is assess the value of information. In order to evaluate how good a certain solution is, the total cost for the routing may be used. The total cost can be calculated with the following equation:

$$\begin{aligned}
 \text{Total cost} &= \text{Charter cost} + \text{Fuel cost} + \text{Delay cost} \\
 &= C_{tot}^{TC} + C_{tot}^F + C_{tot}^D \\
 &= \sum_{k=1}^K C^{TC} x_k + \sum_{i,j=1}^N \sum_{k=1}^K C^F D_{ij} x_{ij} + \sum_{k=1}^K \sum_{i=1}^N C^D x_{ik}
 \end{aligned} \tag{2.1}$$

where

K represent the set of days included in the planning period

N represent the set of customers

C^{TC} represent the time charter cost, calculated based on NOK/day

C^F represents the fuel cost, calculated based on NOK/nm

C^D represents the delay cost related to delivery at each location, calculated based on days with empty storage NOK/day

x_k is equal to 1 if the vessel is chartered at day k

x_{ij} is equal to 1 if the ship sails from location i to j

x_{ik} is equal to 1 if location i is beneath the required inventory limit at day k

3 | Literature Review

As mentioned, the problem at hand includes the construction of a model for routing of a vessel with regards to priority, weather conditions, and customer demand. In order to get an overview of earlier work related to this problem, a literature review is conducted. In the beginning of the literature review, several methods were reviewed, as illustrated in the mindmap found in Appendix A. After an thorough overview was acquired, more specific literature was assessed. Firstly, literature comprising priority and value of information is reviewed. Secondly, I have decided to look into three different approaches for solving the routing problem; exact methods, approximate methods, and rule-based methods. In the following sections, these approaches will be reviewed together with articles that include weather robustness.

3.1 Priority and Value of Information

Haupt (1989) conducted a survey of literature on heuristic priority rule-based job scheduling. He presents a classification, characterization, and evaluation of fundamental priority rules. Job is, in this context, described as tasks that must be conducted in a shop, but can be interpreted to other industries as well. It is normally based on network analysis, flow between service stations, and can be found in several settings. Haupt defines a priority rule that allows the machine to select the next operation, in addition to several other rules that can be used.

Value of information (VoI) can be interpreted to how much more profitable the plans can become if the outcome of the uncertain events is known, Avriel and Williams (1970). The increase in profitability can be defined as the expected value of perfect information. They try to estimate the upper bounds for the expected value of perfect information, which is achieved through stochastic programming.

Wilson (2015) defines VoI as a quantitative method used to estimate the benefit made by investing in a project. He presents a brief introduction to various VoI methods and a step-by-step guide to calculation. The VoI can be calculated analytically or numerically, where both can be used together with decision models. The VoI will, in this thesis, be assessed as a byproduct, and a simplified analysis will, therefore, be conducted.

3.2 Exact Methods

Within the field of exact optimization and routing methods, there exist several typical problem types that can be compared with the problem of the thesis. A popular approach in the oil and gas supply chain is the use of a vehicle routing problem (VRP). Halvorsen-Weare et al. (2012) solve a supply vessel planning

problem (SVPP) for the oil and gas industry, determining the optimal fleet with corresponding weekly routes and schedules. They present a voyage-based solution method for a periodic VRP (PVRP). First, all feasible voyages are generated before the mathematical voyage-based model is solved. According to Statoil, now Equinor, the model proved to be very useful, and they obtained an annual cost saving of about USD 3 million, achieved by reducing the number of supply vessels in their fleet. The model was, on the other hand, not able to solve larger real-life problems within the predefined process time, and it does not include any robustness.

Due to weather affecting the duration of the operations and sailing speed, Statoil experienced delays in their schedules. Halvorsen-Weare and Fagerholt (2011) addressed this problem and tested several approaches for creating robust schedules. One approach was to add some "slack" in the planning process, defined as the supply vessels idle time between voyages. The slack was implemented in the voyage based solution method from Halvorsen-Weare et al. (2012), submitted as a working paper in 2010, by using a robustness profit, so that the model values idle days. They also developed a solution method that combines optimization and simulation in the generation of robust schedules. The simulation utilizes Markov chains to simulate the change in weather state with statistical data from the North Sea. A computational study showed that adding robustness criteria gives a lower predicted cost than if not including it. The effect of the robustness is higher for larger problems and more time windows. This model introduces ways of including robustness in the routing model, something that can be interpreted into the thesis model. The model does, on the other hand, not include the management of the needs at each location nor priority.

Sandvik et al. (2019) try to understand the ship and ship system behaviour in various operational conditions. This is achieved through optimization for simulation of an operational sea passage scenario. The vessel travels from one port to another. During the voyage, an optimization procedure generate a new heading and speed for the ship. It was discovered that a variation in operational scenarios causes a significant difference in required power and fuel consumption. How the author conducted the optimization and included the weather impact, inspire the solution to the thesis problem.

Agra et al. (2017) considers a maritime inventory routing problem (MIRP) for a salmon farmer in Norway. The presented problem includes several concerns related to the thesis problem, such as routing, customer demand, and inventory management, excluding the weather and priority dependency. They present a comprehensive mathematical model previously made by one of the authors, Agostinho Agra, along with some formulation improvements such as special order sets, tightening of time constraints, and dynamic cut generation of clique inequalities. When they used the branch-and-cut algorithm, the real problem was not solved within a reasonable running time. As a response to this, they proposed two matheuristics, one solving the problem as a traveling salesman problem (TSP) and one relaxing the routing variables. One of the matheuristics finds better solutions within a limited amount of time, while the improved mathematical formulation finds feasible solutions for most instances.

Agra et al. (2018) also consider a MIRP, but they include uncertain sailing times due to weather in their model. They assume the routing decisions and the unloading quantities to be fixed, while the time of visit and inventory levels can be adjusted. They propose a decomposition algorithm that iterates between two problems, one considering feasible solutions and one locating infeasible solutions. The decomposition procedure proved to be effective, but by including more robustness the running times

increased, and the problem became harder to solve.

As explained in the previous paragraphs, the MIRP has several similarities with the thesis problem, and the solution methods used to solve it should be further researched with regards to the thesis problem.

3.3 Approximate Methods

Another possible approach for solving this routing problem is with approximate methods such as heuristics. As can be seen in the following paragraphs, there exist several heuristics that have been applied to similar problems.

Fagerholt and Lindstad (2007) has designed an interactive optimization-based decision support system (DSS) for ship routing and scheduling. The DSS utilizes a multi-start local search heuristic to provide quick solutions. The local search heuristic works in a similar manner as with the neighborhood search by working through the search area by applying small local changes. The article also introduces an approach for designing a DSS, something that can be relevant for the thesis problem. The problem handled in the article covers industrial and tramp shipping, but the ideas may be possible to implement in a short-sea transport problem as well.

Fagerholt et al. (2013) presents a mathematical model and a tabu search heuristic to solve a ship routing and scheduling problem in project shipping. The cargoes are here transported on a one-time basis, which complicates the requirements for the storage onboard the ships. The tabu search heuristic proved near-optimal solutions which can give substantial improvements to the current manual planning in the shipping industry.

Borthen et al. (2018) solves an offshore SVPP by introducing a genetic search-based heuristic. In this paper, they use a hybrid genetic search algorithm to minimize the sailing costs by use of a fixed fleet and later minimize the fleet with iterations. The method showed excellent results compared to earlier methods and can be used for large, real-life scenarios. It does not fully capture the complex practical concerns of offshore logistics, such as the effect of bad weather, and the authors, therefore, suggest a multi-objective optimization model as further work.

Kisialiou et al. (2018) presents a large neighborhood search (LNS) heuristic for a periodic supply vessel planning problem (PSVPP). The article covers both fleet composition and scheduling of the vessels. The model made by Halvorsen-Weare et al. (2012) was used as a basis for comparison and problem definition. The presented heuristic improves the generated voyages by removing, reassigning, reducing, and relocating voyages, number of vessels, and visits. When compared to the model by Halvorsen-Weare et al., the heuristic is quite effective for the instances considered and able to solve larger problem instances, which can be practical for the thesis problem.

Kisialiou et al. (2018) tries to account for the uncertain weather conditions at offshore oil and gas installations by developing a methodology for the generation of robust supply vessel schedules, including the trade-off between vessel costs and service level. They focus on the PSVPP, where the objective of the problem is to minimize cost while satisfying the offshore installations' visit requirements. They develop a heuristic algorithm based on an existing adaptive LNS (ALNS) heuristic but with requirements on schedule robustness. They present a new methodology for robust supply vessel routing and scheduling

by introducing a robustness parameter α controlling the duration of the vessel schedule slack. Two kinds of slack are included, inter- and intra-voyage slack, to avoid delays due to uncertain weather conditions. The schedules produced are tested in a simulation model, where the service level of each schedule and the performance under the influence of weather conditions are assessed.

The heuristics used for routing, and especially periodic routing, are as mentioned genetic heuristics, variants of LNS heuristics, and local search heuristics. When compared to the problem at hand, the LNS heuristic has features most likely necessary for the model for a robust priority-based periodic routing.

3.4 Rule-based Methods

The latter approach reviewed is a less popular method, called a rule-based method. To the author's knowledge, this kind of problem has not been solved by practicing this method. In the following paragraphs, relate-able problems or possible solution approaches are therefore reviewed.

Haupt (1989) survey the literature of heuristic priority rule-based job shop scheduling. He looks at the day-to-day job scheduling and shop management decision system, and assess the rules used for job scheduling. One policy that is often included in dynamic job shop scheduling is a priority rule. This rule allows an idle machine to choose its next operation among a list of currently waiting jobs. The priority of the job to be performed depends on if the measures are due-date-based or flow-time-based. Related to this classification, he formalizes a list of basic priority rules which support the decision of the priority of the operations. This kind of practice of deciding priority can be interesting to practice on a priority-based vessel routing.

Kjølleberg (2015) considered a problem similar to the problem at hand, except for the priority of the locations. The author tried to solve the supply vessel weather routing problem (SVWRP) by designing a decision support system (DSS) in MATLAB. This program calculates candidate routes by use of weather forecasts and, together with a ship resistance model, calculate the cost optimal route. The problem is viewed as a TSP where the objective is to find the cheapest route for visiting the oil rigs considering the weather. His work can be used as inspiration for a possible model in MATLAB for the thesis problem, for example, a rule-based method together with simulation.

The rule-based method is an unexplored way of planning the routing of a vessel, but an interesting way of solving problems due to a lack of given constraints and requirements of the method.

3.5 Conclusion of Literature Review

As demonstrated in the previous sections, there exist several possible strategies for solving the thesis problem. The mentioned articles comprise a diverse selection of strategies, ranging from exact methods to heuristics and rule-based methods. The reviewed literature has given the author valuable insight into similar problems, which can be transferred to the thesis problem. Especially the literature about inventory routing in the fish farming industry and weather routing has been valuable. In addition, the literature has introduced constraints, which should be accounted for in the thesis problem as well, such as inventory management and slack. The collection of the various methods have also inspired the author

to look at alternative methods, such as the rule-based method. This method provides the author with the opportunity of being creative and defining the model without constraints.

The experiences from the literature study have led to the decision of further research of three different methods; a mathematical formulation of a MIRP, an ALNS heuristic, and a rule-based method. These approaches will be further explored in the following chapter.

4 | Methodology

In the following chapter, the methods mentioned in Chapter 3 are evaluated and the most promising method chosen. The chosen method is further explained and related theory reviewed.

4.1 Possible Methods

As mentioned in Chapter 3, there exist several possible strategies for solving a priority-based routing problem. I have chosen to look further into three methods; a mathematical formulation of a MIRP solved using branch-and-bound, a large neighborhood search heuristic, and a rule-based method. In the following sections, the chosen methods are further explored and discussed before one method is chosen.

4.1.1 Exact Method

A possible strategy for solving the priority-based routing problem is by use of a mathematical formulation and an exact solution method. The article by Agra et al. (2017) studied in the Literature review introduces a MIRP similar to the thesis problem. A salmon farmer has to ensure that the feed at the factory and the fish farms is well within the inventory limits. They have two cargo vessels available and the possibility of importing feed from an external source, both of which are not relevant in this case. They present a mathematical model for the problem, consisting of an objective function and four groups of constraints; routing, loading and unloading, inventory, and timing. All of these constraints would be included in a mathematical model for the thesis problem, in addition to the weather dependency and location priorities. A description of the objective and constraints that a possible mathematical model would include is presented in Table 4.1.

Their model is solved using a branch-and-bound algorithm and two matheuristics, in which a branch-and-bound technique would be interesting to test. The branch and bound technique has been applied with success to many operations research problems, (Hillier and Lieberman, 2015). The basic concept underlying this method is to divide and conquer. The problem is divided into smaller and smaller problems, which is subsequently solved. For further explanation of the Branch-and-bound technique, the reader is referred to Hillier and Lieberman (2015), Chapter 12.6.

The mathematical model by Agra et al. (2017) was demonstrated to be too big to solve within a reasonable running time using the branch-and-bound algorithm. When including two more groups of constraints, weather and priority, the model will probably become even more complicated. The fact that the thesis problem only has three possible locations to sail to and only one vessel is believed to simplify

the model, but not necessarily enough. A mathematical model is still a structured way of presenting the problem, and it is relatively easy to include the necessary constraints. An overview of the needed objective and constraint groups are presented in the following figure.

Objective/constraint	Description
Objective function	Minimize fuel cost, charter cost and delay cost
Routing constraints	Flow conservation, correct end location, all locations visited, re-routing option, binary constraint of decision variables
Loading and unloading constraints	Load on ship before and after loading/unloading, ship capacity, capacity at location, amount of need changed at each location, non-negativity requirements
Inventory management	Inventory levels adjusted for change, storage capacity, boundary level of inventory, feasible inventory level at end of planning horizon, meet customer demand
Timing	Relate start and end time of visits, consistency in timing of visits, keep visits within planning horizon, constraints at port
Weather dependency	Operational limits, availability at site, relation between locations, possibility of waiting for weather
Priority	Locations with higher priority should be prioritized before those with lower if inventory levels are within limits

Table 4.1: Description of objective function and constraints relevant for a mathematical formulation of the thesis problem.

To conclude, a MIRP is most likely able to include the constraints of the problem, but the problem may be too large for the branch-and-bound technique to handle, Agra et al. (2017). Agra, therefore, suggest a matheuristic as a possible solution. A matheuristic is the combination of mathematical programming and a metaheuristic. The matheuristic applied by Agra, A. et al. use the mathematical model as a basis, and is based on improving already existing feasible solutions by fixing variables in an iterative procedure, (Agra et al., 2017). A heuristic is able to solve more extensive problems, as will be discussed in the next section.

4.1.2 Large Neighborhood Search (LNS)

As mentioned in the Literature survey, the LNS metaheuristic has become a popular method for planning problems in the offshore industry. One of the articles presented, the robust supply vessel scheduling by Kisialiou et al. (2018), used an existing ALNS heuristic together with robustness criteria to generate robust schedules. All LNS algorithms build on the experience that by searching a large neighborhood, local optima with high quality can be found, (Pisinger and Ropke, 2010). A neighborhood in this context is solutions nearby the current solution, defined by a selected type of heuristics, for example, 2-opt

neighborhood or the relocate neighborhood. An initial solution is improved by alternately destroying and repairing the solution, as is illustrated in Figure 4.1.

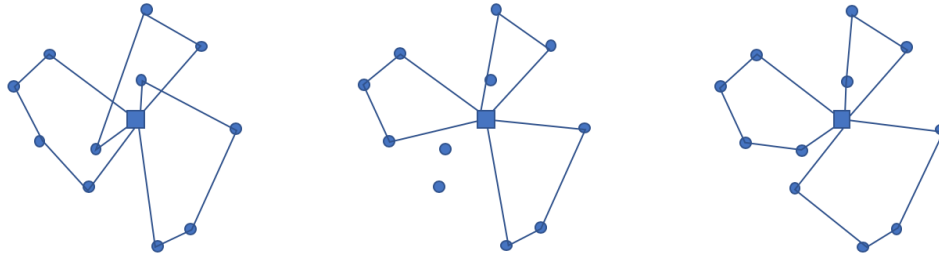


Figure 4.1: Illustration of the principle of destroy and repair, with inspiration from (Pisinger and Ropke, 2010). The left figure shows a VRP solution before the destroy operation and the middle figure after 3 customers are removed. The right figure show the solution after the repair operation.

An ALNS heuristic is an extension of the LNS heuristic that allows the use of multiple destroy and repair methods within the same search, (Pisinger and Ropke, 2010). The different destroy and repair methods are assigned weights that manage the use of each method. The weights are updated during the search process. In this way, the heuristic is always updated with regards to the state of the search and the problem at hand. A pseudocode for the ALNS can be seen in Algorithm 1, (Pisinger and Ropke, 2010).

Algorithm 1: Pseudocode for the adaptive large neighborhood search method.

- 1: input: a feasible solution x
 - 2: $x^{best} = x$
 - 3: ρ^-, ρ^+ = variables that store the weight of each destroy and repair method
 - 4: **repeat**
 - 5: select destroy and repair methods within set of methods using ρ^- and ρ^+
 - 6: choose new solution x^{new} in neighborhood of x^{best}
 - 7: **if** x^{new} is feasible **then**
 - 8: $x = x^{new}$
 - 9: **end if**
 - 10: **if** cost of $x^{new} <$ cost of x^{best} **then**
 - 11: $x^{best} = x^{new}$
 - 12: **end if**
 - 13: update weights of methods
 - 14: **until** stop criterion is met
 - 15: **return** best solution x^{best}
-

Because Kisialiou et al. (2018) designed a robust model including several of the factors related to the problem at hand, an ALNS heuristic appears to be a feasible solution method. The only thing that must be included is the priority of each location and inventory management, which is considered to be manageable. The problem with the ALNS is that it can be time-consuming if the problem is considered too big, but for the simple case, it is assumed to be acceptable. A more critical problem is the fact that the

this problem is not task-restricted but consumption-restricted. The need at each location is dependent on time, and the time between each visit will, therefore, have consequences for the degree of demand and the subsequent visits. This may complicate an ALNS based model.

4.1.3 Rule-based Method

A rule-based method in this context is a tailored model build from the ground and up. It can be recognized as a type of heuristic, with specific rules that control the model. It has the possibility of being build with requested preferences and accuracy, and it does, in theory, not have to follow any given guidelines. In this way, the model can be tailored to the problem at hand, and everything from routing and inventory management to weather and priority dependency can be included. It is also believed that a rule-based method will be easier to implement in the MATLAB optimization environment together with a simulation compared to the other methods.

Possible disadvantages with a rule-based method are that we do not know anything about its run time. As far as the author knows, a model like this has not been made for this problem type before, and we, therefore, do not know the model type. It could be a time-consuming method, both with respect to the generation of the model as well as the run time, and it can be uncomplicated. The fact that the problem consists of a single vessel routing with only three different locations simplifies the routing problem drastically, and a commercial solver is presumably not necessary. The model should regardless be built as general as possible, in order to have the possibility of model expansion.

In order to portray a possible rule-based model for routing of priority-based operations, pseudocode for an imaginable model is illustrated in Algorithm 2.

Algorithm 2: Pseudocode for the rule-based routing algorithm.

- 1: **input** weather forecast for the next three weeks and start/current inventory levels
 - 2: generate overview of available locations with regard to weather conditions and operational limit
 - 3: generate prediction of inventory level for the next week
 - 4: make empty list of operations
 - 5: **for all** days within planning period **do**
 - 6: **for all** locations **do**
 - 7: fill in list with time and location to visits based on chosen rule set
 - 8: **end for**
 - 9: **end for**
 - 10: **output** routing plan for vessel containing time and location for the next three weeks
-

As presented in Algorithm 2, a list is generated based on a chosen rule set. This rule set must be generated based on the desired focus of the routing. For example, a rule set may have the focus of ensuring the inventory levels are well within satisfactory levels before a period of bad weather, or they can focus on minimizing the number of visits at each location.

Because the choice of rules is free, such kind of a model can become too general to be used, or it can introduce a new way of routing a vessel between locations with different exposure and needs.

4.1.4 Choice of Method

When comparing all three methods reviewed, as seen in Table 4.2, the ALNS and the rule-based method stand out as the most promising methods. The MIRP is believed to be hard to model and be time demanding. The ALNS algorithm is proved to be effective in a similar problem, but it is uncertain how it will be compatible with consumption-based routing instead of a task-restricted routing. The ALNS is also believed to be less flexible compare to the others. Especially the Rule-based method is believed to be flexible, since the rules and model can be determined by the person in charge. The fact that a rule-based heuristic for this kind of problem has not, as far as the author knows, been tried before compose an interesting challenge. I, therefore, choose to use a rule-based model when solving the priority-based routing problem at hand.

	MIRP	ALNS	Rule-based
Include all constraints	Yes	Maybe	Yes
Difficulty level	Hard	Easy	Unknown
Time	Long	Short	Short
Consumption-restricted	Yes	No	Yes
Flexibility	Yes	No	Yes

Table 4.2: Comparison of possible methods.

4.2 Rule Set Formulation

As concluded in the previous paragraph, a rule-based model is chosen to solve the priority-based routing problem. In order to be able to make such a model, the factors deciding where the vessel should go must be further explored before effective rules can be made. This is achieved by looking at three cases where the decisions of the captain are analyzed.

4.2.1 Exploring the Problem

As mentioned in Section 2, several factors can influence the choice of route, as illustrated in Figure 4.2. The inventory level represents the urgency of a visit. The weather conditions, together with the operational limit of the vessel, decide whether the location can be visited now or in the future. The priority of the location represents the importance of delivering to that location instead of others. The distance affects the time of the operation, and the fuel cost and the charter cost influence the choice of numerous visits. The factors both complicate the problem, in addition to composing the opportunity of more efficient solutions.

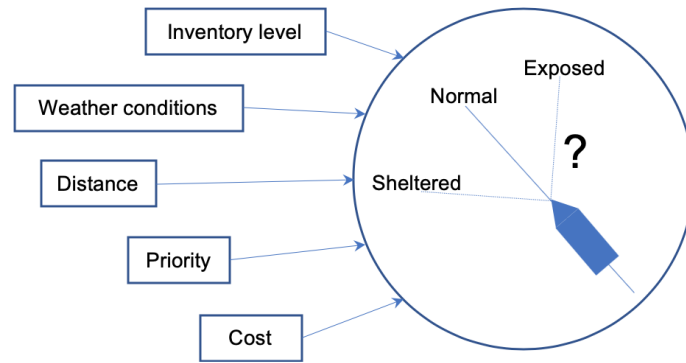


Figure 4.2: Factors influencing the choice of route.

In order to understand the problem dynamics and logic, the factors running the model must be further explored. The relationship between the factors must be identified and further explored, which is achieved through several case scenarios performed manually in Excel, which will be explained in the following sections, Case A, B and C.

Case A

In the first case, Case A, three factors; weather conditions, priority, and inventory level, are reviewed. The purpose of the case is to evaluate the importance of each factor and discover how complex the problem dynamics are. In this case, all the related costs for each location are assumed to be equal. It is also assumed that a location with full inventory storage cannot be visited. The distances are scaled in consistency with the problem description, with the exposed location being further away from port than the normal and sheltered location. It is assumed that only one location can be visited each day, as illustrated in Figure 4.3.

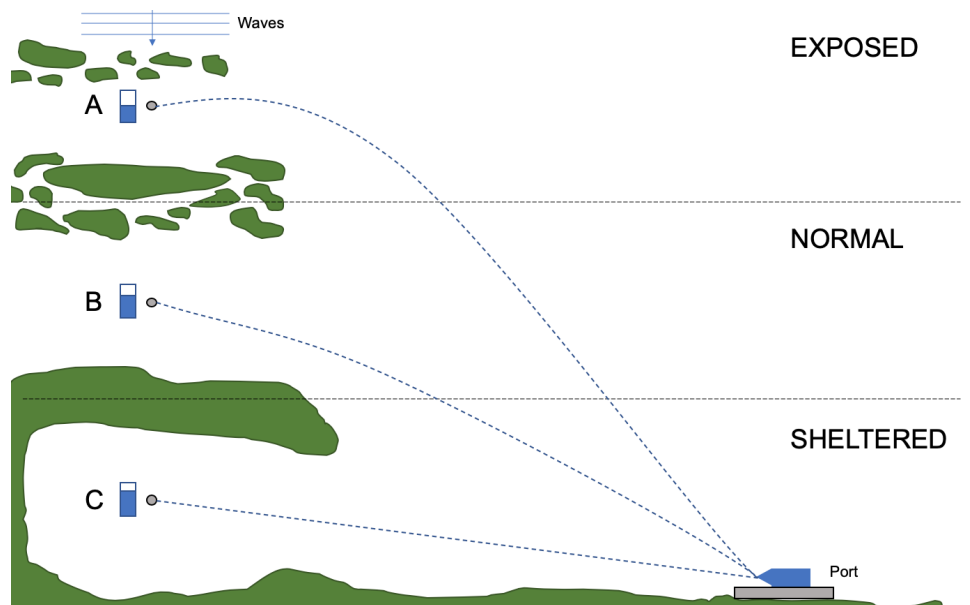


Figure 4.3: Illustration of simple model used in manual case analysis.

The first case, Case A, is constructed as simple as possible in order to focus on the logic of the decisions made. Three different locations are assessed; one exposed, one with normal exposure, and one sheltered. They have identical inventory capacity and must be visited so that the inventory level does not go empty. Location A correspond to the exposed location, location B, the normal, and location C the sheltered location, as seen in Figure 4.3. It is assumed that the vessel only travels from port to one location and then back to port. The distance to the site and the time needed for the operation are not included in the case. The conditions for the case can be seen in Table 4.3.

	A - Exposed	B - Normal	C - Sheltered
Priority	High	Medium	Low
Weather conditions	Low, Medium, High	Low, Medium, High	Low, Medium, High
Inventory level	Low, Medium, High	Low, Medium, High	Low, Medium, High

Table 4.3: Conditions for Case A scenarios.

As can be seen in Table 4.3, the only variables in the test scenario are the inventory level, and weather conditions at the different locations; low, medium, and high, and different combinations of those are therefore assessed. The priorities are kept constant, as stated in the problem definition. By applying a systematic approach with the conditions in Table 4.3, a wide variety of combinations are assessed, as can be seen in Appendix B. The inventory levels and the weather conditions for each location are compared, before a choice for which location to visit is made. This choice is made based on inventory level, weather conditions, or priority. An illustration of how the test is completed can be seen in Figure 4.4. When there is no need for the vessel to sail, it stays in port, as seen for Scenario 1. In Scenario 15, when all locations have a medium inventory level, the priority decides where the vessel should sail, and the vessel, therefore, sails to location A because it has the highest priority. In Scenario 23, the low inventory level at the sheltered location determines the choice of location.

As a result of Case A, it is experienced that the inventory level is the dominant factor for the decision of where to go. The inventory level was the most commonly used basis for the choice of location. The second dominant factor was the priority; in other words, the probability of bad weather that delay operation. The priority of the location was dominant in the decision process when there was a tie with regards to inventory level or if the weather condition is at medium and it is a possibility that a period with bad weather is arriving. To conclude, the inventory level is the dominant factor, but the priority controls the decision when there are several possibilities.

Scenario	A - EXPOSED			B - NORMAL			C - SHELTERED			My choice:	Reason
	Inventory level	Weather conditions	Priority	Inventory level	Weather conditions	Priority	Inventory level	Weather conditions	Priority		
1	High	Good	High	High	Good	Medium	High	Good	Low	-	Stay in port
15	Medium	Good	High	Medium	Good	Medium	Medium	Good	Low	A	Inventory + priority.
23	Medium	Good	High	High	Good	Medium	Low	Good	Low	C	Inventory.

Figure 4.4: Example of Test A procedure.

The second case, Case B, consider a weekly based scenario where the weather forecast for the next

week is known and the captain can make decisions based on this. This case is more detailed and includes exact numbers for operational limits, inventory capacity, and inventory consumption, as seen in Table 4.4. The inventory consumption is assumed to be constant and similar for all locations so that the future need of each location can be calculated. The choice of the next site to visit is made day by day based on a known weather forecast one week ahead and inventory levels. The inventory levels in the Excel document are coded to reduce in conjunction with the day rate, except when they are visited by the vessel, and the storage is filled up. An assorted collection of start inventory levels between 0 and 2000 are tested, as can be seen in Appendix C.

	A - Exposed	B - Normal	C - Sheltered
Priority	High	Medium	Low
Inventory capacity	2000	2000	2000
Inventory consumption [/day]	300	300	300
Availability	40%	70%	Always

Table 4.4: Conditions for Case B scenarios.

The wave heights used in the case are historical wave data from *waveclimate.com*. The wave height at 06:00 each day for five weeks is used as a basis for the weather condition. The same weather data is used for all locations. The availability at each location is set to be 40% for the exposed and 70% for the normal, while the sheltered location is always available. The availability for the exposed and the normal locations are not realistic as they are very low, but this is intentional in order to force decisions with regards to weather. The priorities are equal to the priorities in Case A. Each test is started at random inventory levels between 0-2000, and the weather forecast is only known one week in advance. The capacity of the vessel is assumed to be large enough to fill up the inventory at the locations regardless of the level at the site. The days are considered separately, but with the weather forecast for the next week in mind. A decision of where to go next is based on weather conditions, inventory, or priority. An example of the procedure in Test B can be seen in Figure 4.5.

Week	Day	Hs [m]	A - Exposed		B - Normal		C - Sheltered		Next stop:	Reason:
			Inventory level	Availability	Inventory level	Availability	Inventory level	Availability		
1	Monday	1,18	1700	1	1700	1	1700	1	B	Inventory + Weather
1	Tuesday	1,86	1400	1	2000	1	1400	1	A	Inventory + Weather
1	Wednesday	2,72	2000	0	1700	1	1100	1	C	Inventory
1	Thursday	4,09	1700	0	1400	0	2000	1	-	No place to go
1	Friday	2,61	1400	0	1100	1	1700	1	B	Inventory
1	Saturday	2,91	1100	0	2000	1	1400	1	C	Inventory
1	Sunday	3,5	800	0	1700	0	2000	1	-	No place to go

Figure 4.5: Example of Test B procedure.

In Figure 4.5, the availability with regards to weather conditions is visualized with a color code. Green represents that the location can be visited, while red represents that the location is not available.

The availability is scaled for each location, based on the exposure. In this case, a plan for the next week should be generated based on the given weather forecast. On Monday the next stop is chosen as B because we want to go to A right before the period with bad weather arrives. The weather forecast indicated that the bad weather would be present for an extended period of time so that the inventory level should be as full as possible when the weather arrives in case future visits are not possible. On Wednesday, location C is chosen due to a low inventory level. It should be pointed out that the choices made for the next stop in these scenarios are the personal choices of the author.

As a result of Case B, the importance of planning is identified. If a location does not have a full inventory level when a long period of bad weather arrives, it is a high likelihood that the inventory storage will go below the lower limit, and a delay cost must be charged. Another experience that was made is that if the vessel does not have a minimum amount of cargo it should deliver, it ended up sailing several trips with minimal amounts of cargo. A minimum limit for transport should, therefore, be included, and if none of the locations require a visit, the vessel should stay in port to save cost. Lastly, there should be made specific limits for the inventory levels in order to classify them as low or high. Specific limits make the decision process easier.

The last case analyzed, Case C, includes the time and cost of the operations in the routing decisions. Eight scenarios are presented, covering possible tricky situations the vessel might experience. Various availability, inventory levels, costs, and cost relationship are tested. The documentation for the test can be found in Appendix D. An example of a scenario tested is low inventory levels at all locations, good weather at all locations except the normal location, which experiences somewhat harsh weather conditions. The delay cost is based on priority, with the highest delay cost for the exposed location. In this scenario, it was chosen to sail to the normal location first before a potential bad weather period and then sail to the exposed location. The sheltered location should be visited the next day since it has the lowest delay cost. This order is based on total cost due to the delay cost and the priority of the locations. The overall experience made from Case C is that if the delay cost is differentiated, it will dominate in the decision process. The delay cost would then be a representation of the priority, but that is not the intention. The delay cost for each of the locations is therefore chosen to be equal.

In conclusion, Case A, B, and C provided the author with several learning experiences that should be taken into consideration during the modeling. As mentioned, low inventory levels are dominant as long as no longer periods with bad weather is predicted. The priority of the locations was experienced to make a difference in the cases where there is a tie or bad weather periods arrive. It should also be set a lower limit on the amount of inventory the vessel can deliver in order to avoid numerous visits with low amounts.

4.2.2 Proposing Rule Sets

As stated in Algorithm 2, a rule set must be chosen for the routing of the vessel. A rule set is a collection of specific rules that must be followed and which subsequently generate a plan for the vessel. As mentioned, it is desired to look at the value of information and the value of more complicated rule sets with regards to routing. In order to do so, 4 rule sets are developed based on the factors explored in Section 2.3. As described, inventory levels and weather conditions contribute to the routing process, but in order to know how much, rule sets with various information are tested. The rules included in the rule sets are based on the experiences made from the test scenarios explained in Section 4.2.1 and further exploration of the problem.

Rule set	Information given
Rule set 1	Last visit Consumption rate Storage capacities
Rule set 2	Same as Rule set 1 + Weather forecast for the next day
Rule set 3	Same as Rule set 1 + Weather forecast for the next 2 weeks
Rule set 4	Same as Rule set 3 + Inventory levels at each location

Table 4.5: Description of information available in each rule set

It is desired to look at the effect of including more information in the decision process. This is achieved by first only knowing the storage volumes and consumption rates. For each rule set, portions of added information are given, such as weather forecast for the next day and subsequently the next weeks, as seen in Table 4.5. Lastly, information about storage levels is given. All rule sets allow for at most two operations during one voyage. During the scheduling of operations, the priority of the locations should always be considered.

In Rule set 1, described in Algorithm 3, very little information is known, only the consumption rate at the facilities, the storage capacity at each location and the vessel, and when the last visit was conducted. These parameters are imported before the old plan is reviewed and the last time the location was visited is found.

In Rule set 1, a visit is booked if there have been 4 days since the last visit. Four days are chosen based on the consumption rate and the inventory capacity, which implies that a location lasts 6.67 days withing a visit. If a location is unavailable at the day of operation, the operation is canceled and deleted, and no rescheduling is done. A slack of 4 days is therefore included. The model is then able to book a visit the next day based on last visit, since the visit was deleted. Because it is possible to visit two locations during one voyage, it is not known how much supplies are unloaded at each location.

Algorithm 3: Pseudocode Rule set 1.

```
1: import needed parameters
2: find last day each location is visited
3: for all days left of planning period do
4:   review exposed, normal, sheltered
5:   if days since last visit > 4 then
6:     book visit in plan
7:     update last day visited
8:   end if
9:   increase day value by one
10: end for
11: if location not available then
12:   visit is cancelled
13: end if
```

In the second rule set, information about the weather forecast for the next day is also given in addition to the information given for Rule set 1. If an operation according to the weather forecast cannot be conducted, it is now possible to reschedule the operation to the next day instead of canceling, as shown in Algorithm 4. Other than that, the visits are booked in the same manner as in Rule set 1.

Algorithm 4: Pseudocode Rule set 2.

```
1: import needed parameters
2: find last day each day is visited
3: import weather forecast for next day
4: for all days left of planning period do
5:   review exposed, normal, sheltered
6:   if days since last visit > 4 then
7:     book visit in plan
8:     update last day visited
9:   end if
10:   increase day value by one
11: end for
12: if location not available then
13:   visit is moved to next day if available
14: end if
```

In Rule set 3, Algorithm 5, additional information about the weather forecast is given. Here the weather forecast for the next 13 days is included in the information provided. The weather forecast can be fetched from a weather matrix, as will be described later. The weather forecast and the operational limit of the vessel is used to calculate the availability of each location. This availability is checked when an operation is booked. If an operation is canceled due to an unavailable location, this rule set should manage to reschedule the operation the following day.

Algorithm 5: Pseudocode Rule set 3.

```
1: import needed parameters
2: find last day each day is visited
3: import weather forecast for next 2 weeks
4: for all days left of planning period do
5:   review exposed, normal, sheltered
6:   if days since last visit > 4 then
7:     if location available then
8:       book visit in plan
9:       update last day visited
10:    else
11:      no visit is booked
12:    end if
13:  end if
14:  increase day value by one
15: end for
```

Algorithm 6: Pseudocode Rule set 4.

```
1: import parameters, inventory levels
2: import weather forecast for next day
3: make inventory level prediction
4: for all days left of planning period do
5:   review exposed, normal, sheltered
6:   if predicted inventory level < lower limit
       then
7:     if location available then
8:       book visit in plan
9:       update predicted inventory levels
10:    end if
11:  end if
12:  increase day value by one
13: end for
```

In the last rule set, Rule set 4, information about the current inventory levels at each location is also provided. Based on these levels, predictions for the future inventory levels can be made and visits booked based on these. As indicated in Algorithm 6, the visits to each location is booked based on when the predicted inventory levels reach a specific lower limit. Since the rule set also has information about the weather, the availability is calculated and checked before booking a visit.

A visit cannot be booked or completed if the location is not available, something the rule set must control. The rule sets are built in MATLAB and consist of several IF- and FOR-loops. The MATLAB-scripts for each rule set can be found in Appendix E.

4.3 Exploring Uncertainty through Simulation

In order to compare the performance of each rule set it, a simulation should be conducted. The case scenarios assessed in Section 4.2.1 are based on a perfect weather forecast where the weather is assumed to become as predicted. In order to test how the rule sets handle uncertainty, uncertainty must be included in the simulation. This is achieved by testing the rule sets in a year-long simulation of the vessel operation. A schedule for the vessel is generated based on the chosen rule set, and this plan is followed by the simulation until the weather forecast has changed, or the end of the plan has been reached, and a

new plan must be generated.

By introducing uncertainty into the model, it is possible to test the rule sets up against each other. This can be achieved by introducing several degrees of uncertainty into the model. In the beginning, the rule sets can be exposed to perfect weather conditions where operations may always be conducted. If the rule set conquer nice weather, they can be tested with uncertain weather to test the robustness. If they manage to produce satisfactory plans with a perfect weather forecast, next thing they are exposed to the weather with some uncertainty. To further separate the rule sets, a higher degree of uncertainty may be included. In order to test the rule sets with the use of an uncertain weather forecast, a simulation model in Simulink is made.

4.3.1 Simulink and SimEvents

Simulink is a block diagram environment inside MATLAB that can be used for multidomain simulation, (MathWorks, b). MATLAB is integrated into the environment so that MATLAB algorithms can be incorporated and simulation results exported. A simulation model in Simulink can be used to test any desired system under chosen conditions, which can reduce the overall cost of a project.

The system in question, a vessel performing several operations based on a plan that should be generated at certain moments, can be described as an event-driven system. Once the vessel has performed the desired task, and the plan is finished, a new plan must be generated. In order to simulate such a system, discrete-event simulation should be used. Discrete-event simulation models the system in question as a string of events, where one event triggers the next event. Inside Simulink it is a tool for such kind of systems called SimEvents. It can be used to model event-driven systems using blocks from a given component library, (MathWorks, a). It is widely used in operational research, routing, scheduling, and supply-chain management.

4.3.2 Weather Representation

The plan generated with the rule sets may be affected by the weather conditions at different locations. In order to simulate the weather conditions, historical weather data in the form of significant wave height, H_s , is downloaded. This data can be used to generate realistic weather forecast. A realistic weather forecast is similar to the values found in the historical data, where the change in wave height from one hour to the next is rational. It is in other words not possible to go from $H_s = 10\text{m}$ at 06:00 to $H_s = 0.5\text{m}$ at 07:00. The wave heights must be correlated. In order to generate such a weather forecast, two methods are evaluated. One method fit the weather data to a probability distribution and draw random values. These values will be realistic since they will follow the probabilities of each sea state, but the following values will not be correlated. An alternative method is by use of Markov Chains, where the next value is only dependent on the current, and each transition is based on calculated probabilities. Since real-life weather is correlated, Markov chains are chosen as the method for weather generation.

4.3.3 Markov Chain Simulation

A realistic weather forecast can be generated using Markov chains. A Markov chain is a stochastic process where the next state only depends on the last state; the model has no memory. The transition between one state and another is based on certain probabilistic rules that can be calculated. A state in a Markov chain can represent a sea state. In order to describe the principle of a Markov chain, an example will be described. Figure 4.6 illustrate a Markov chain transition diagram between three sea states. The values on the arrows represent the likelihood of transitioning from one state to another. When the current sea state is 2, there is a 0.4 likelihood that it will transition to sea state 1, 0.1 to sea state 3, and a 0.5 likelihood it will remain the same. The transition diagram is the visual representation of a transition matrix, as seen in Table 4.6.

	Calm	Moderate	Rough
Calm	0.5	0.4	0.1
Moderate	0.4	0.4	0.1
Rough	0.1	0.4	0.5

Table 4.6: Example of a Markov chain transition matrix.

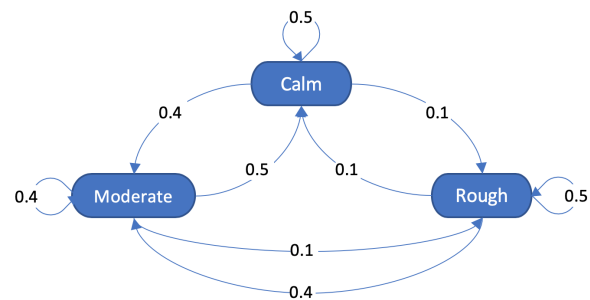


Figure 4.6: Markov chain transition diagram.

In the following paragraph, a Markov chain will be described in more detail. We start with a state s chosen from a set of states $S = s_1, s_2, \dots, s_r$, College. The process moves successively from one state s_i to another, s_j . The probability of moving from s_i to s_j is denoted p_{ij} . This probability only depends on the current state, not any of the previous. p_{ij} is called the transition probability, the probability of transitioning from s_i to s_j . This probability is given for all possible outcomes, including the probability of remaining in the current state, p_{ii} . A transition matrix is a matrix containing all the transition probabilities, p_{ij} .

4.4 Value of Information

Value of Information (VoI) is as mentioned in Section 3 a concept within decision analysis. VoI can be described as the value of reducing uncertainty by excersising data collection, Wilson (2015). It can therefore be used as a method for assessing the cost effectiveness of future projects. VoI is based on statistical decision theory, where an objective function should be maximized and the choice of course of action leads to uncertain payoffs. The uncertainty of the payoffs can be reduced by investing time and money into further research of the problem and gaining access to more information. According to Wilson (2015), the main question in a VoI analysis is "Does the value of additional information outweigh its cost?".

A VoI analysis can be conducted with either analytical or numerical methods. The methods presented by Wilson (2015) assess the potential payoffs, where they use values such as expected value of perfect

information and expected value of sample information to calculate an expected net payoff and loss. These analyses are based on using sampled data with probabilities for various course of actions.

In this thesis, a simplified analytical method will be conducted. Instead of using probability and assessing different outcomes, the total cost for each rule set will be calculated based on various simulations and then compared. The rule sets are, as mentioned, generated based on various levels of information. Hence, the value of information will be able to assess based on the total cost for each rule set.

In order to access this type of extra information, an investment must be made, such as constructing a weather buoy to log metocean conditions or a system for controlling the inventory levels. These investments will not be further explored, but discussed when the total costs are evaluated.

5 | Testing Rule Sets with Simulation

This chapter describes how the rule sets are implemented in the model, and the simulation model used to test the rule sets.

5.1 Model Setup

The model generate a plan based on the chosen rule set and tests the method in a simulation model in Simulink. A flow chart for the model can be seen in Figure 5.1. At simulation start, the necessary input is imported into a MATLAB script, which generates a new plan for the vessel. This plan is exported to the simulation model in Simulink where the vessel carries out the plan under the influence of the weather. A new plan is generated if one of two incidents happen: the end of the plan is reached, or the updated weather forecast deviate sufficiently from the old. The simulation continues for a year before it is terminated, and the simulation output is presented. Each part of the model is described in the following sections.

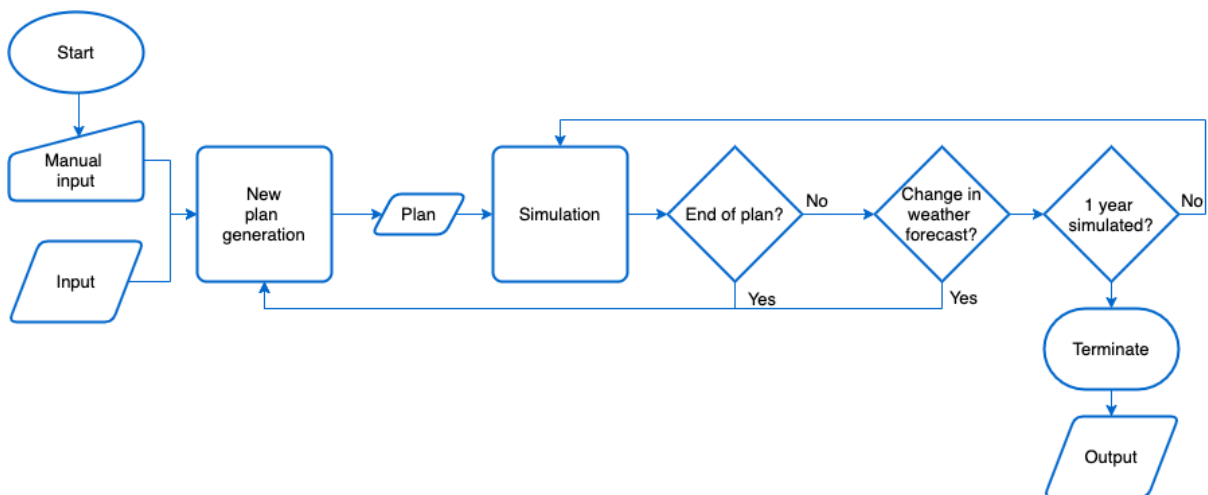


Figure 5.1: Flow chart for the overall model.

5.2 Start of Model

The MATLAB script *run_simulation.m* start the model. This script import the necessary parameters, distances, and attributes into the MATLAB workspace available for use both inside MATLAB and Simulink.

This script also generates the real weather conditions for one year and the availability for the first two weeks, as will be explained in the next paragraphs.

5.3 Model Input

The model input consists of manual input and input imported from various data sources. The manual input controls the degree of uncertainty in the weather forecast and the seed number. In addition, the model import all the parameters and numbers needed to generate a new schedule for the vessel. Certain parameters are constant and, therefore, only imported once, while others are variables and must be imported every time a new plan is generated. The imported input includes the vessel characteristics and other parameters, the weather conditions, and the rule sets.

5.3.1 Vessel Characteristics and Parameters

Three characteristics define the vessel used in the model; speed, cargo capacity, and operational limit, as seen in Table 5.1. The symbol # illustrates that the number may be changed manually if desired. The values are imported from an Excel document called *ShipAttributes.xlsx* by use of the MATLAB command *xlsread* and saved in the MATLAB workspace. When imported, it is saved as a 3x1-matrix.

Speed	Cargo capacity	Operational limit
# [kn]	# [tons]	# [m]

Table 5.1: Ship attributes.

Other parameters that must be imported to MATLAB are the various cost rates and inventory parameters. These are read from an Excel document called *Parameters.xlsx* and saved in the MATLAB workspace as a matrix to reduce simulation time. The Excel file contains the parameters presented in Table 5.2. The values can be modified by changing the value in the Excel document.

Parameter	Value	Unit
Start inventory levels	#	[ton]
Consumption rate	#	[ton/day]
Lower inventory limits	#	[ton]
Delay costs	#	[NOK/day]
Fuel cost	#	[NOK/nm]
Charter cost	#	[NOK/day]

Table 5.2: Cost and inventory parameters.

5.3.2 Weather Conditions

In order to have a relation between the weather forecast and the real weather, a large weather matrix is made using the MATLAB script *WeatherMatrixOneYear.m*. This script imports the chosen historical weather data with significant wave heights and generates the corresponding transition matrix. The weather data is analyzed and divided into eight sea states with a state range of 1.09 m. The transition between the sea states is saved in a transition matrix T, which is used to generate the real weather vector.

A matrix for the real weather is generated by preallocating a matrix with the size 380x14. 380 days are used in order to get the correct weather forecast for day 365 as well. The rows in the matrix correspond to the day of the simulation, while the columns represent the current weather and the weather forecast for the next 13 days for that given day. Column one is the real weather vector generated. The weather forecast is based on real weather. An illustration of how the weather matrix is built can be seen in Figure 5.2.

		Weather forecast Next 13 days					
		This day	Next day	Day after	...	Last day	
Days with corresponding weather forecast	Day 1	Day 1	Day 2	Day 3	Day 4	...	Day 14
	Day 2	Day 2	Day 3	Day 4	Day 5	...	Day 15
	Day 3	Day 3	Day 4	Day 5	Day 6	...	Day 16
	:	:	:	:	:	:	:
	Day 365	Day 365	Day 366	Day 367	Day 380
	:	:	:	:	:	:	:
	Day 380	Day 380					

Figure 5.2: Illustration of weather matrix.

The values in the weather forecast depend on the degree of weather uncertainty chosen when the model is run. Four degrees can be chosen, zero, low, medium, or high. Zero generates perfect weather, where all locations are always available. Low uncertainty indicates a perfect weather forecast, and the weather forecast is therefore set equal to the real-life weather in column one. Medium and high weather uncertainty indicates that the weather forecast is generated based on the real weather the next 13 days, but with an included uncertainty. Random numbers generated by MATLAB decide whether the sea state will stay constant or change. If a change is chosen, a new random value decides if the sea state increase or decrease. The likelihood of a change of sea state and the size of the change increases further into the future. In other words, the weather forecast is more precise closer to the current day. A weather forecast is generated for each day in the simulation period, in total 14 days.

The matrix with the real weather and weather forecasts is preprocessed before the simulation and imported to the simulation model. This is conducted to save simulation time.

5.3.3 Rule Set Implementation and Plan Generation

The simulation model is used to test the various rule sets. In order to do so, a specific rule set must be chosen for each run. The sets are saved as various scripts with their associated names, such as *Ruleset1.m*. Before each simulation, the desired rule set must be chosen.

The rule sets inside MATLAB generate a plan for the first two weeks. An example of a plan can be seen in Table 5.3. Firstly, a 14x4 matrix with zeros is preallocated for the plan. The first column represents the day of operation, which will be between 1-365. The second column represents the location the vessel should visit if a visit is booked. 1 means the exposed location, 2 the normal, and 3 the sheltered location. 0 indicates that the vessel should stay in port. The third column includes the possibility of visiting a second location during one voyage. If two locations demand a visit, it is possible to sail to two different locations during one voyage. The last column describes if the visit has been completed, 1 if completed, 0 if not. In Table 5.3, the visits until day 7 have been completed, while the rest is not. Visits to the various locations are scheduled based on several IF- and FOR-loops fulfilling the rules of the chosen rule set. The rule sets are generate based on the pseudocodes illustrated in Algorithm 3, 4, 5 and 6. The MATLAB scripts for the various rule sets can be found in Appendix E. After a plan is generated, it is imported to Simulink.

Day	Location	Double	Completed
1	1	0	1
2	3	0	1
3	2	0	1
4	0	0	0
5	0	0	0
6	0	0	0
7	1	0	1
8	0	0	0
9	3	0	0
10	2	0	0
11	0	0	0
12	0	0	0
13	1	0	0
14	0	0	0

Table 5.3: Format of the plan generated.

5.4 Simulation Model

Once the weather matrix and the first plan is generated, the simulation model inside Simulink is entered. An illustration of the routing model can be seen in Figure 5.3. It is a discrete event simulation model built up by SimEvent blocks. The simulation stop time is set to 8760 hours, which corresponds to one year. One time unit in simulation time, therefore, corresponds to one hour. In the following paragraphs, the central parts of the simulation model will be presented and explained.

First, the vessel entity is generated in the vessel generator. The vessel then sails to the port where the plan is reviewed. If a visit is booked, the entity sails to the next stop. In the "Sail to assigned location", the vessel goes to the planned location to complete the operation. After this, the vessel should go to the port to review the plan again. If no visit is booked, the vessel waits 24 hours before it reviews the plan again.

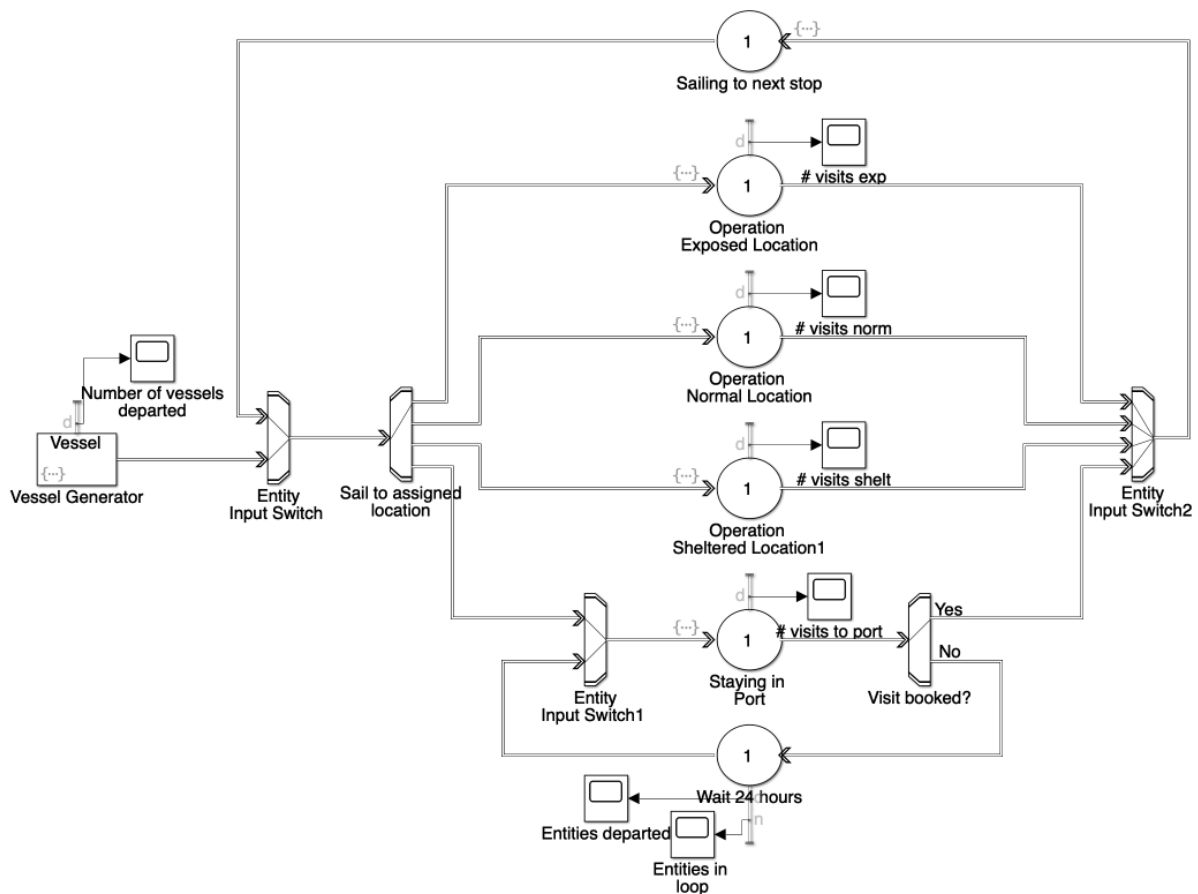


Figure 5.3: Snapshot from Simulink model illustrating the four different places to go; exposed location, normal location, sheltered location and port.

5.4.1 Entity Generation

The entity flowing through the model is a vessel entity. The vessel is generated in the Vessel generator. A scope is used to verify that the correct number of vessels is generated, in this case, one. As mentioned, the vessel has certain characteristics. These characteristics are imported from the MATLAB workspace and assigned as attributes to the vessel entity generated inside the Event Actions to the Vessel Generator. In addition to the imported attributes, the vessel has other attributes used to navigate inside the model. A list of all the attributes assigned to the vessel entity can be seen in Table 5.4. How and where they are used in the model will be explained at the relevant blocks.

Nr.	Attribute Name	Attribute Initial Value
1	Speed	#
2	CargoCapacity	#
3	OperationalLimit	#
4	NextStop	1
5	TimeToNextStop	1
6	Double	1
7	LastStop	1
8	Switch	1
9	LastDay	1

Table 5.4: Vessel attributes with initial value.

The vessel generation is completed at time zero, and is, therefore, the first thing that is performed inside the simulation model. In order to follow the plan generated in MATLAB, it is imported from the MATLAB workspace and saved in the global variable *plan*. What a global variable is and how it is defined is explained in the following section.

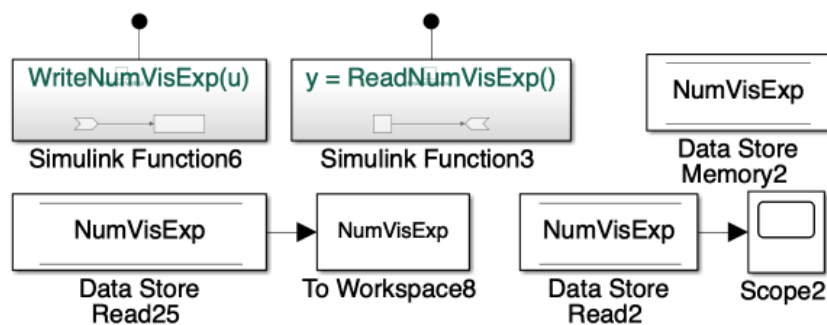


Figure 5.4: Illustration of the blocks used to define a global variable.

5.4.2 Global Variables

Global variables can be accessed from the *Event actions* inside any block in Simulink. A global variable is generated by defining a Data Store Memory as the variable in question, for example *NumVisExp* which

counts the number of visits to the exposed location, see Figure 5.4. The number can be updated by generating a Simulink Function called *WriteNumVisExp(u)* containing a Data Store Write block named the same as the Data Store Memory, *NumVisExp*, which write the value *u* to the Data Store. In order to read how many visits that have been conducted, another Simulink Function must be generated, called *y = ReadNumVisExp()*. Inside this function, the value from the Data Store for the *NumVisExp* will be read and set equal to the desired variable. By calling for both of these functions inside the Event Actions to the Exposed Operation Server, the values for *NumVisExp* can be updated every time the vessel enters the exposed location. This procedure is used for all global variables in the model. An overview of the global variables in the model with descriptions can be seen in Table 5.5.

Name	Size	Unit	Description
TotDelayCost	3x1	[NOK]	Matrix containing total separate delay cost at exposed, normal and sheltered location
CharterCost	1x1	[NOK]	Total charter cost calculated
FuelCost	1x1	[NOK]	Fuel cost
NumVisExp	1x1	[-]	Number of visits to the exposed location
NumVisNorm	1x1	[-]	Number of visits to the normal location
NumVisShel	1x1	[-]	Number of visits to the sheltered location
NumDaysPort	1x1	[-]	Number of days the vessel stay in port
NumNewPlan	1x1	[-]	Number of times a new plan is generated
InvLevExp	1x1	[ton]	Inventory level at exposed location
InvLevNorm	1x1	[ton]	Inventory level at normal location
InvLevShel	1x1	[ton]	Inventory level at sheltered location
AmountUnloaded	1x1	[ton]	Amount unloaded from ship per day
Time	1x1	[hours]	Simulation time, where 1 = 1 hour
Day	1x1	[day]	Keep track of days, 1 Day = 24 hours
plan	14x4	[-]	Schedule for the next two weeks containing when to visit where and if the visit has been completed

Table 5.5: Description of all global variables in the model.

5.4.3 Routing

The vessel entity flow through a round trip model where it can choose between four different places to go to; exposed location, normal location, sheltered location or port. At the simulation start, the vessel is forced to go to port, the block seen in Figure 5.5. The vessel should start each simulation from port. In port, the vessel storage is filled up, and the entity attribute *NextStop* is assigned a value from 1 to 4. 1 is the exposed location, 2 the normal, 3 the sheltered, and 4 stay in port. If the attribute *NextStop* is assigned a value, another attribute value, *Switch*, is set to 1. Based on the *Switch* value, the Entity Output Switch called "Visit booked?" decides if a visit should be conducted or if the vessel should stay in port and wait for the next day. If no visits are booked, the entity attribute *Switch* is set to 2, and the vessel

stays in the Wait 24 hours server before entering port again and checking if a visit is booked at the next day. Before leaving port, the attribute *LastStop* is set to 4, which corresponds to port. This value is used to calculate distances traveled for fuel cost calculations.

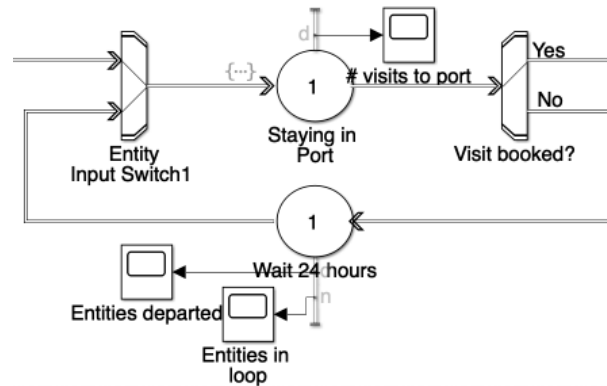


Figure 5.5: Port block in simulation model.

If the vessel is booked for a visit, the *Switch* value is set to 1, and the vessel sails the distance calculated in the server *Sailing to next stop*. The distance is calculated based on the *LastStop* and *NextStop* attributes in addition to the matrix *Distances* imported from the MATLAB workspace. This matrix includes the distances between all of the four locations. The distance sailed is used to add sailing costs to the global variable *FuelCost*. The global variable *CharterCost* is also updated in port every time the vessel travels to port.

After the vessel has sailed the intended distance, it enters the "Sail to assigned location". The *NextStop* attribute is used to decide where the vessel should go. If the vessel is booked to the exposed location, the global variable that counts the number of visits, *NumVisExp*, is updated, as previously explained. The inventory level, the global variable *InvLevExp*, is increased with the intended amount, depending on the amount unloaded from the ship. After the unloading, the numbers are updated, both for the vessel and the inventory level at the location. The *NextStop* value is then set to 4, port, unless the global variable *Double* has a value. The *Double* attribute is set to 1,2 or 3 if a second visit is booked on the same day. If that is the case, the *NextStop* is set equal to the *Double* value when the first visit is completed. The same procedure is conducted at all three locations, exposed, normal, and sheltered.

5.4.4 Re-Routing

While the simulation model follows the *plan*, a MATLAB function checks the weather forecast for the current day, see Figure 5.6. The *plan* is generated based on a weather forecast for a given day. The MATLAB function compares the "old" weather forecast, the forecast used to generate the current plan, with an updated one for the current day. If the updated weather forecast shows a reduced availability to the exposed or the normal location, a new plan must be generated.

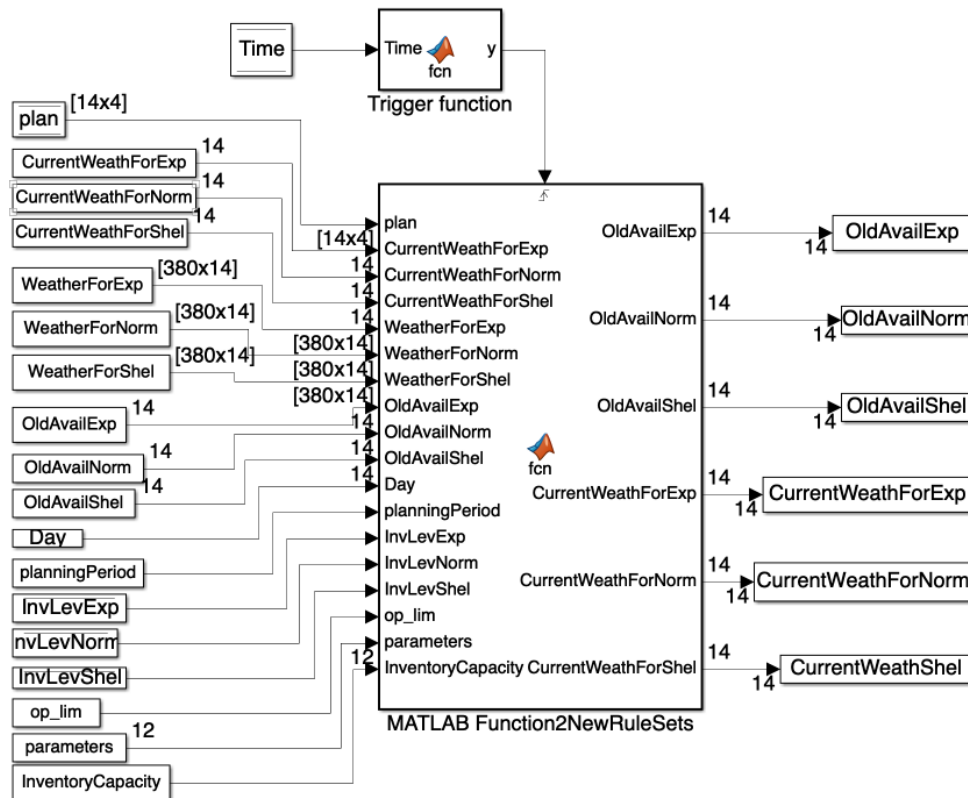


Figure 5.6: Illustration of the MATLAB function used for the rerouting and generation of new plan.

The MATLAB function use the correct weather forecast from the weather matrix, *WeatherForExp*, calculate the availability at each location with the updated forecast and compare it with the availability for the old forecast, *OldAvailExp*. If the availability has changed, or the plan is completed, a new plan is generated inside the MATLAB function using the same rule set as in MATLAB. The new plan is then saved to the global variable *plan*. The current weather forecast used in the plan generation is saved to the MATLAB workspace as *CurrentWeathForExp* so that it can be compared with the future updated weather forecast. The availability is also saved in the MATLAB workspace as *OldAvailExp*.

5.5 Model Output

Once one year has been simulated, the global variables updated during simulation is sent back to MATLAB. In the *run_simulation* script, desirable numbers, graphs, and bars are made with the simulation output, as will be illustrated in the Case Study.

5.6 Verification of Model

During the design of the model, various blocks were used to verify that the model performed as intended. The *Scope*- block was widely used to control the movements of the entities and the change in the global variables. If three visits were planned to the Exposed location, a scope illustrating the number of departures from the Exposed server was used to verify that three visits were conducted.

In order to control the generation of the plan, a *Display*- block was used. This block shows the current value of the global variable *plan* at each time instance. The plan was also output in the command window inside MATLAB during simulation to show the change.

Other methods that were used to verify that the model worked as intended were the use of debugging inside MATLAB and leaving out the semicolon. The debugging was a great tool for problem assessment and checking if the model did what it was supposed to do. The semicolon is used to suppress the output of a code line. If it is left out, it outputs this value, which proved to very helpful during the model setup.

6 | Case Study

In order to test the rule sets generated and assess the value of information, four case studies are conducted. In the following sections, the case details and parameters will be explained before the results of the case studies are presented and discussed. The case studies that are conducted test all the four rule sets with different uncertainties in weather and different seeds for the random numbers.

6.1 Case Description

The case study involves feed distribution to three different aquaculture facilities in the area around Frohavet next to Frøya and Hitra, two islands off the coast of Norway. The facilities are placed in areas with different exposure, as seen in Figure 6.1. Location A is located in an exposed area in Frohavet, and location B is located near Sistranda next to Frøya with normal exposure while location C is located near Fillan at Hitra in a sheltered area. The port is located northeast on the island Hitra at a place called Sandstad.



Figure 6.1: Map over aquaculture facilities in case study. Map from Kystverket (2020).

The case involves the planning and scheduling of feed distribution to the three locations. The four case studies vary in terms of the degree of uncertainty included in the planning process. Case 1 plans for perfect weather and does not include any form of uncertainty. Case 2 includes a perfect weather forecast

where the weather will be as predicted. Case 3 includes some uncertainty in the weather forecast. Case 4 include a very uncertain weather forecast. Four levels of uncertainty are studied in order to assess the robustness of the rule sets.

	Weather Type	Degree of Uncertainty
Case 1	Perfect weather	Zero
Case 2	Perfect weather forecast	Low
Case 3	Uncertain weather forecast	Medium
Case 4	Very uncertain weather forecast	High

Table 6.1: Description of cases.

6.2 Case Details

The distances between each location are real and can be seen in to the left in Table 6.2. They are calculated using the distance calculator in Google Maps, Google (2020). The chosen characteristics of the vessel can be seen to the right in Table 6.2. The values are based on the characteristics of typical short sea shipping vessels, in exception to the operational limit, which will be explained. The other parameters used in the case can be seen in Table 6.3 in the next page.

	A	B	C	Port	Characteristic	Value	Unit
A	0	15	21	23	Speed	10	knots
B		0	9	16	Cargo capacity	3000	tons
C			0	7	Operational limit	4	sea state
Port				0			

Table 6.2: Left: Overview of distances between the different locations in nautical miles. Right: Ship characteristics.

Parameter	Value	Unit
Start inventory level A	1700	ton
Start inventory level B	1400	ton
Start inventory level C	1100	ton
Consumption rate	300	ton/day
Lower inventory limit A	600	ton
Lower inventory limit B	600	ton
Lower inventory limit C	600	ton
Delay cost A	5000	NOK/day
Delay cost B	5000	NOK/day
Delay cost C	5000	NOK/day
Fuel cost	1000	NOK/nm
Charter cost	10 000	NOK/day

Table 6.3: Parameter values in case study.

The initial inventory levels for each location are chosen to be different from each other. The consumption rate is set to 300 tons/per, which means that the storage level will reach zero after 6.67 days when the storage capacity is 2000 tons. The lower inventory limits are used in Rule set 4 and are chosen to be equal for all locations, but has the possibility of being changed. The same applies to the delay costs.

6.3 Weather Data

Historical weather data is used to generate the transition matrix, which generates real weather. The weather data consist of the significant wave height with combined wind waves and swell for the coordinates 63.93N 9.08E, which correspond to the exposed location in Frohavet. The data is downloaded from Copernicus (2019) as a netcdf file where it is logged every third hour each day for 10 years. The data is handled in MATLAB by reading the file using the command `ncread()`. The probability distribution of the weather data used can be seen in Figure 6.2. The data is divided into eight sea states with equally sized bins. The ranges and the probability of each of the sea states can be seen in Table 6.4.

Sea state	Range	Probability
1	0 - 1.085 m	0.31
2	1.085 m - 2.17 m	0.46
3	2.17 m - 3.256 m	0.15
4	3.256 m - 4.341 m	0.054
5	4.341 m - 5.426 m	0.015
6	5.426 m - 6.511 m	0.0040
7	6.511 m - 7.596 m	0.0012
8	7.596 m - 8.681 m	0.00033

Table 6.4: Sea state ranges and their associated probabilities.

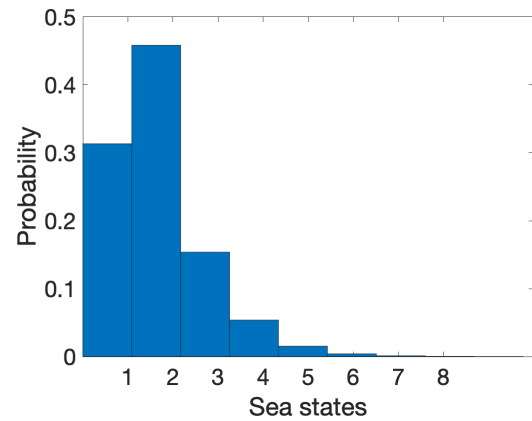


Figure 6.2: Probability distribution of the H_s values included in the data set.

As mentioned in the problem description, a location can be classified as exposed if the probability of the location being unavailable is more than 20%. Based on this and the distribution of H_s in Figure 6.2, an operational limit is set to sea state 3. The calculated likelihood of a sea state 3 or higher is 22.8%. Normally, the operational limit is a H_s value given for the specific vessel, but in this case, the value for the vessel is set based on the weather data to satisfy the problem description. The transition matrix based on the weather data can be seen in Table 6.5. This matrix is used to generate the weather used in the simulation.

Sea state	1	2	3	4	5	6	7	8
1	0,925	0,075	0,001	0,000	0,000	0,000	0,000	0,000
2	0,052	0,904	0,044	0,001	0,000	0,000	0,000	0,000
3	0,000	0,134	0,796	0,067	0,002	0,000	0,000	0,000
4	0,001	0,001	0,202	0,708	0,084	0,005	0,000	0,000
5	0,000	0,000	0,000	0,336	0,576	0,080	0,008	0,000
6	0,000	0,000	0,000	0,000	0,398	0,490	0,112	0,000
7	0,000	0,000	0,000	0,000	0,034	0,448	0,310	0,207
8	0,000	0,000	0,000	0,000	0,000	0,000	0,667	0,333

Table 6.5: Transition matrix generated based on historical data.

6.3.1 Various Realizations

In order to test the rule sets with various realizations of weather, three various seeds are used for generating the weather conditions at various locations. For each case, all the rule sets are tested with three different seeds, the number that controls the random number generator in MATLAB, made with pseudo-random series of numbers. The use of seeds also makes sure that each simulation with the same seed, experience the same random numbers, which is practical when comparing various rule sets that are tested with the same degree of uncertainty.

Seed 1 is equal to 12345, Seed 2 represents 23456, and Seed 3 represents 34567. Each seed generates

various weather conditions. For the exposed and the normal locations, the weather conditions may affect the availability, while the sheltered location is always available. The following figures illustrate the sea states at the exposed and the normal location for seed 1, 2, and 3. The red line represents the operational limit of the vessel.

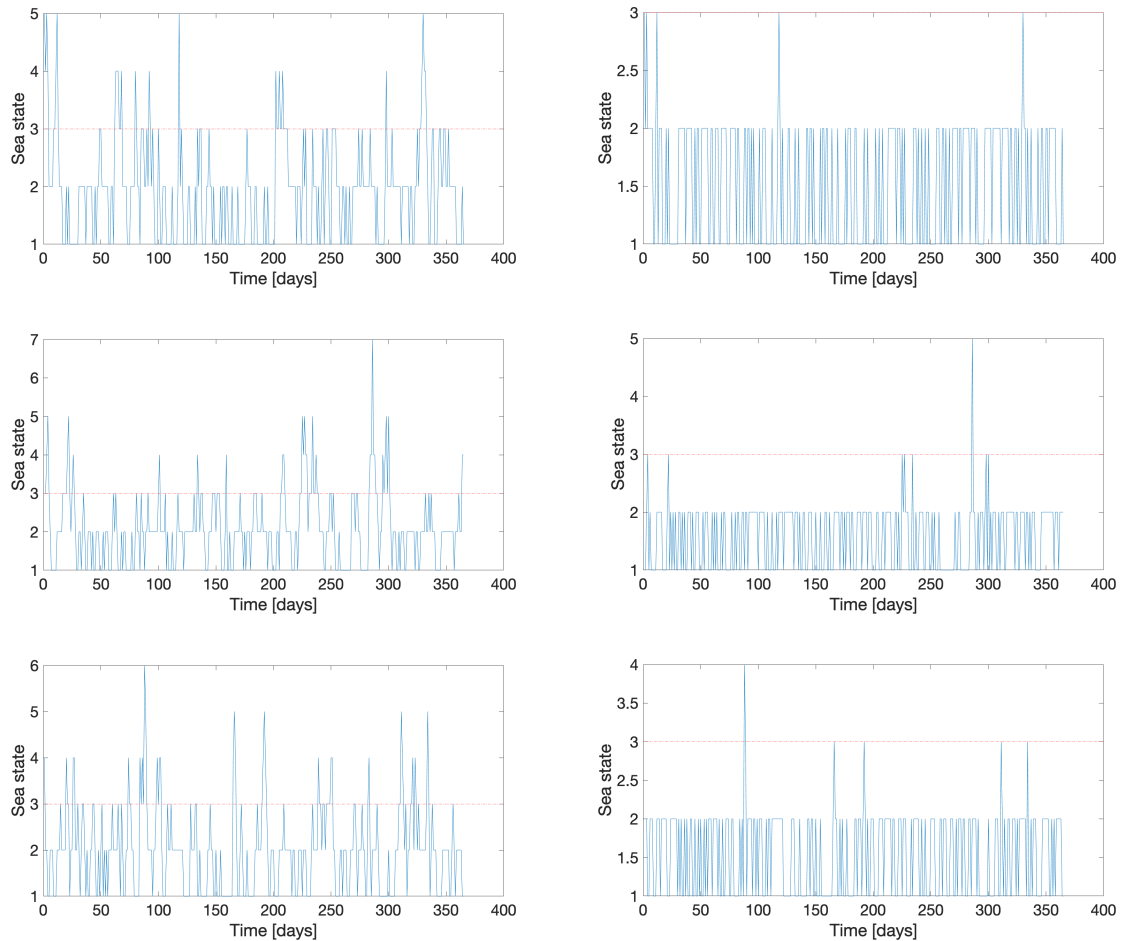


Figure 6.3: Sea states for exposed in column one and normal in column two. Row one is for Seed 1, row two for Seed 2 and row three for Seed 3.

As can be seen the pictures in the first column in Figure 6.3, the exposed location experience several periods with bad weather. The normal location only experiences one occasion of bad weather for each seed, as seen in the second column. Since the weather at the normal location is scaled from the exposed, the harsh weather periods at the normal location occur concurrently.

6.4 Results from Simulation and Evaluation of Rule Sets

Each rule set is tested with four cases, with three various seeds for each case, which in total represents 48 simulations. In the following sections, selected results from the simulation are presented. Firstly, an overall summary of the total cost is presented before more detailed results for each case, and rule set is provided and discussed.

6.4.1 Total Cost

As mentioned in Section 2.4, the total cost is the sum of the charter, fuel, and delay cost. In Table 6.6, the total cost in NOK for all 48 simulations are presented. The values in each cell represent the values for the Seed 1, 2, and 3, and their average value in italics.

	Rule set 1	Rule set 2	Rule set 3	Rule set 4
Case 1	8,096,000	8,096,000	8,096,000	8,333,000
	8,096,000	8,096,000	8,096,000	8,333,000
	8,096,000	8,096,000	8,096,000	8,333,000
<i>Average</i>	<i>8,096,000</i>	<i>8,096,000</i>	<i>8,096,000</i>	<i>8,333,000</i>
Case 2	8,615,000	8,642,000	7,997,000	7,229,000
	8,572,000	9,400,000	8,471,000	7,042,000
	9,588,000	9,262,000	9,594,000	7,125,000
<i>Average</i>	<i>8,925,000</i>	<i>9,101,333</i>	<i>8,687,333</i>	<i>7,132,000</i>
Case 3	8,686,000	8,707,000	7,886,000	6,583,000
	8,800,000	8,987,000	8,364,000	7,235,000
	9,363,000	9,527,000	9,392,000	7,526,000
<i>Average</i>	<i>8,949,667</i>	<i>9,073,667</i>	<i>8,547,333</i>	<i>7,114,667</i>
Case 4	8,728,000	8,863,000	8,406,000	6,979,000
	9,113,000	9,055,000	8,778,000	7,347,000
	8,722,000	9,368,000	7,698,000	7,212,000
<i>Average</i>	<i>8,854,333</i>	<i>9,095,333</i>	<i>8,294,000</i>	<i>7,179,333</i>

Table 6.6: Summary of the total cost in NOK for each simulation. The four values in each cell are from simulations with Seed 1, 2 and 3, and their average value.

As indicated in Table 6.6, in Case 1, the total costs for Rule set 1, 2, and 3 are identical. This is as predicted since the only difference between them is the information about the weather forecast, and the weather is, in this case, perfect and none of the locations are unavailable. The cost with Rule set 4 is calculated to be slightly higher due to more days in operation and, therefore a higher charter cost. This will be discussed later.

In Case 2, where the weather forecast is perfect, the total costs increase for Rule set 1, 2 and 3, while it is reduced for Rule set 4. Rule set 4 knows the inventory levels and the weather and can, therefore,

route the vessel based on this. The difference in cost between Case 2 and 3 is low for all rule sets. This could indicate that it does not matter if the weather forecast is known for two weeks ahead instead of just one day. Case 4, with the highest uncertainty, also has relatively small changes in the total costs for the various rule sets, which is not as predicted.

Overall, Rule set 4 handles the uncertain conditions in Case 4 the best. Compared to Rule set 1, 2, and 3, Rule set 4 generates a total cost that is around 1 million NOK cheaper than with Rule set 3.

6.4.2 Simulation Output

In the following tables, a summary of the simulation output for each rule set and case is presented. The results presented here are from simulations with Seed 2. The simulation output for all 48 simulations can be found in Appendix H.

Rule set 1

In Table 6.7, a summary of the simulation output in MATLAB for Rule set 1 with Seed 2 is presented. Rule set 1 generates plans based on consumption rate and storage capacity; it does not know the weather conditions, the inventory level at each location, or the amount delivered during each visit. If the location is unavailable, the visit is cancelled.

Rule set 1	Case 1	Case 2	Case 3	Case 4
Total cost [NOK]	8,096,000	8,572,000	8,800,000	9,113,000
Charter cost [NOK]	1,840,000	1,940,000	2,040,000	2,080,000
Fuel cost [NOK]	6,256,000	6,512,000	6,675,000	6,938,000
Delay cost [NOK]	0	120,000	85,000	95,000
Total number of visits	276	263	261	271
Days in operation	183	193	203	207
Days in port	182	172	162	158
Visits to exposed location	92	77	75	81
Visits to normal location	92	94	93	98
Visits to sheltered location	92	92	93	92
Number of new plans generated	26	206	208	266
Run time	38 sec	34 sec	42 sec	36 sec

Table 6.7: Summary of simulation results for Rule set 1, Case 1, 2, 3 and 4 with Seed 2.

The average total cost with Rule set 1 increases with the case number, except for a small reduction in Case 4, as seen in Table 6.6. The same trend can be seen in Table 6.7, except for the reduction in cost in Case 4. With Seed 2, the total cost is increased, while the delay cost is reduced, which also can be reflected in the number of days the vessel is in operation. The number of days the vessel is in use increases with the uncertainty. In Case 4, the vessel completed in average 1.3 visits per day in operation. Since the vessel has the ability to do 2 visits each day, it has a potential for improvement.

The difference between Case 3 and 4 is the level of uncertainty. A change in sea state is achieved by adding or subtracting a value dependent on the day in question and a random number from a normal distribution, as described in 5.3.2. In Case 3, this value is divided by 10, while in Case 4 it is divided by 5, as presented in the MATLAB code for the weather matrix in Appendix G.4. In this way, the sea state in Case 4 can in theory receive a bigger change in value compared to Case 3, but it does not necessarily mean that it will always be higher. The normal distribution cover values from the complete solution space, 0-1, but the variable often hold values close to the expected value. Due to the randomness in the values presented, it can, therefore, happen that Case 3 stands out as more uncertain than Case 4. This can cause a difference in the total cost due to misleading availability, which can be the case for both Case 2, 3, and 4. If the random value reduces the sea state for the following day so that it is within the operational limit, the model believes that the location is available and book a visit. When the simulation reaches that specific day, the real weather will be compared to the plan, and the booked visit will not be carried out, which can lead to delay costs.

The number of visits to each location is around 92 for the normal and sheltered location for all cases, while the exposed location has a reduction in visits once weather is included. This is as predicted. The number of visits to the locations are as predicted, since they are booked every fourth day. Since each facility has enough food to last 6.67 days, it would be possible to try to book every fifth day, so that the total number of visits each year is around 73. This might induce an even higher delay cost, as seen in Figure 6.4, or it might save fuel cost with fewer visits.

As seen in Figure 6.4, the delay cost stands for the lowest component in the total cost. As predicted, the exposed location induces most of the delay costs. This is also the case for the other seeds and cases. The exposed location experiences some unavailability, and a delay cost is therefore predicted. The fuel cost is high, which is normal, but the relationship between the costs depends on the rates used.

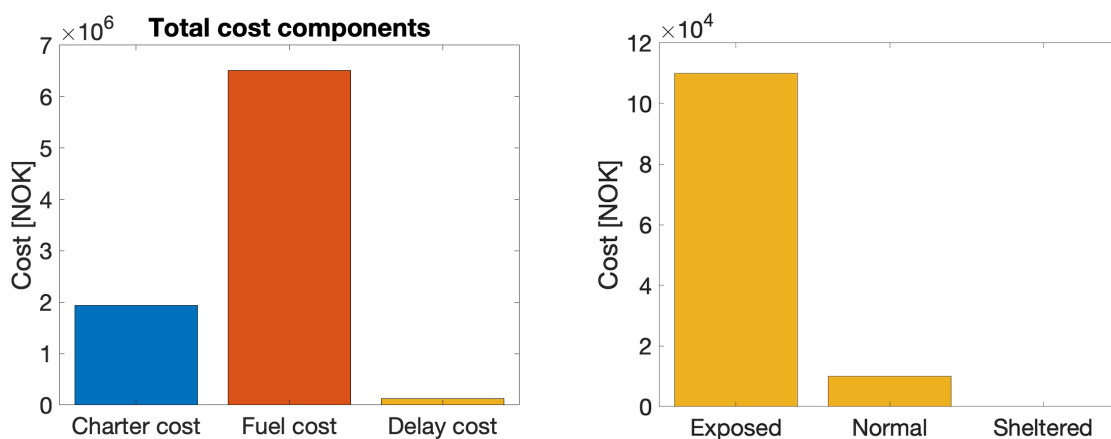


Figure 6.4: Total cost and delay cost for each of the locations for Rule set 1, Case 2, Seed 2.

The number of new plans generated increases drastically from Case 1 to Case 2, due to the inclusion of a weather forecast. In Case 1, a new plan is only generated when the two-week plan is completed. In Case 2, 3, and 4, a new plan is generated when the availability is changed in addition to when a plan is completed. If plan generation costs money, it would be beneficial to lower the number of new plan

generations. The number of new plans generated is the same for all Rule sets for the same Seed. This is because the same weather and start inventory levels are applied. A new plan is generated every time the availability is changed, even though weather is not included in the information given. See Discussion for a further description.

Rule set 2

In Table 6.8, a summary of the simulation output for Rule set 2, and Seed 2 is presented. Rule set 2 is based on Rule set 1, but now the weather forecast for the next day is known. If a visit is booked and the location is unavailable, the visit can be moved to the following day.

Rule set 2	Case 1	Case 2	Case 3	Case 4
Total cost [NOK]	8,096,000	9,400,000	8,987,000	9,055,000
Charter cost [NOK]	1,840,000	2,180,000	2,010,000	2,080,000
Fuel cost [NOK]	6,256,000	7,120,000	6,912,000	6,885,000
Delay cost [NOK]	0	100,000	65,000	90,000
Total number of visits	276	263	266	270
Days in operation	183	217	200	207
Days in port	182	148	165	158
Visits to exposed location	92	85	86	84
Visits to normal location	92	91	92	95
Visits to sheltered location	92	87	88	91
Number of new plans generated	26	206	208	266
Run time	37 sec	36 sec	33 sec	43 sec

Table 6.8: Summary of simulation results for Rule set 2, Case 1, 2, 3 and 4 with Seed 2.

The total cost with Rule set 2 in Case 1 is equal to the cost with Rule set 1, which is as predicted. Rule set 1, 2 and 3 are all based on the time for the last visit, and they should, therefore, generate the same costs during perfect weather conditions. The cost increases with around 1 million NOK from Case 1 to Case 2, which is due to a high increase in fuel and charter cost, as presented in Table 6.8. These costs are related to the increased number of days in operation. Rule set 4 in average complete 1.3 visits per day of operation, which is the same as for Rule set 1. Rule set 2 is therefore not much more efficient than Rule set 1.

Even though the vessel operated more often compared to in Case 1, a delay cost is still obtained. As presented in Figure 6.5, half of the delay cost is from the exposed location, while the normal and sheltered constitute about one fourth each. Delay costs at the exposed location are not desired but predicted since this is the location with periods with unavailability. The fact that the sheltered location also experienced delays exhibit a bad strategy for planning. The delay is probably due to the option of moving an operation if the location is unavailable, and the other booked operation has a lower priority. The sheltered location has the lowest priority so that both the normal and the exposed location can wipe out the visits to the

sheltered location.

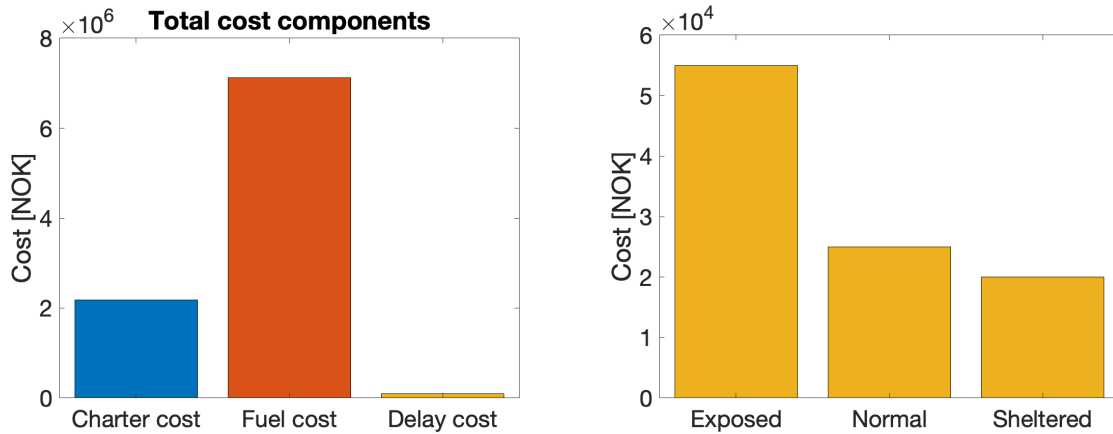


Figure 6.5: Total cost and delay cost for each of the locations for Rule set 2, Case 2, Seed 2.

The days the vessel is in operation increases from Case 1 to Case 2 but decreases slightly in Case 3 and 4. The increase is due to the introduced uncertainty, while the decrease in Case 3 and 4 might be due to weather. If only the total cost and days in operation is considered, it may seem like the rule set handles uncertainty well. But, the goal is to avoid empty storage, and this method is the only one that generates delay cost at the sheltered location. This shows that not only the total cost must be assessed. An empty storage might induce future losses as well.

Rule set 3

Table 6.9 summarize the simulation results for Rule set 3 with Seed 2. Rule set 3 know the weather forecast for the next 13 days in addition to the information given for Rule set 2 and can, therefore, reroute the vessel if the weather forecast change.

Rule set 3	Case 1	Case 2	Case 3	Case 4
Total cost [NOK]	8,096,000	8,471,000	8,364,000	8,778,000
Charter cost [NOK]	1,840,000	1,960,000	1,890,000	2,030,000
Fuel cost [NOK]	6,256,000	6,316,000	6,289,000	6,538,000
Delay cost [NOK]	0	195,000	185,000	210,000
Total number of visits	276	251	251	254
Days in operation	183	195	188	202
Days in port	182	170	177	163
Visits to exposed location	92	65	67	65
Visits to normal location	92	94	93	97
Visits to sheltered location	92	92	91	92
Number of new plans generated	26	206	208	266
Run time	42 sec	52 sec	38 sec	41 sec

Table 6.9: Summary of simulation results for Rule set 3, Case 1, 2, 3 and 4 with Seed 2.

As presented in Table 6.6, Rule set 3 generate the same total costs in Case 1 as Rule set 1 and 2 as predicted. In Case 2, the average total cost increases with about half a million before it decreases in Case 3 and 4. This is not as predicted, as the cost was predicted to increase with uncertainty. This may be the effect of knowing the weather forecast for the next two weeks, and therefore being able to plan, but is somewhat strange that the total cost decreases as much as it does. In Table 6.9, this cost reduction is not evident. After reviewing the simulation output in Appendix H, it is clear that the cost components vary greatly with the seed number. In Case 3, the delay cost varies from 70 000 NOK to 535,000 NOK, while the fuel cost varies with 2 million. The values are, therefore, strongly affected by the seed, which affects how the routing is performed.

At the same time, the cost components are also correlated. When the delay cost is high, the charter cost is low due to fewer visits. When the charter cost is high, the delay cost is low because then most visits are completed. To conclude, Rule set 3 seems to generate better plans when subjected to uncertainty in weather conditions.

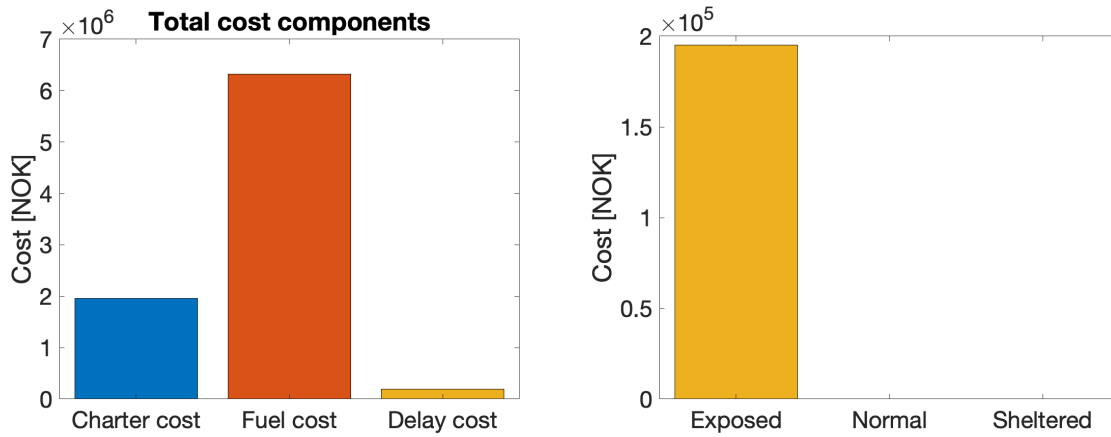


Figure 6.6: Delay cost for each of the locations for Rule set 4, Case 2.

The delay cost only comes from the exposed location, as illustrated in Figure 6.6. The delay cost for Rule set 3 is more than double the delay cost for Rule set 2, as seen in Table 6.8, which is not as predicted since the amount of information given is increased. This reflects the number of visits to the exposed location, which is reduced by 20 compared to Rule set 2, while the number of visits to the normal and sheltered locations are quite similar. This can be explained by the difference in how the weather forecast is used. In Rule set 2, a visit can be moved to the next day if the location is unavailable. In Rule set 3, the visit is canceled, but the rule set should be able to book a new visit the following day before this would be the case. The utilization of the vessel with regards to number of visits per day of operation is the same as for Rule set 1 and 2 with 1.3 visits per day.

Rule set 4

The simulation output for Rule set 4 is presented in Table 6.10. In addition to the consumption rate, storage capacity and weather conditions, Rule set 4 also knows the inventory level at all times for all facilities.

Rule set 4	Case 1	Case 2	Case 3	Case 4
Total cost [NOK]	8,333,000	7,042,000	7,235,000	7,347,000
Charter cost [NOK]	2,040,000	1,570,000	1,700,000	1,690,000
Fuel cost [NOK]	6,293,000	5,177,000	5,245,000	5,407,000
Delay cost [NOK]	0	295,000	290,000	250,000
Total number of visits	205	196	195	198
Days in operation	203	156	169	168
Days in port	162	209	196	197
Visits to exposed location	69	59	57	57
Visits to normal location	68	67	67	69
Visits to sheltered location	68	70	71	72
Number of new plans generated	26	206	208	266
Run time	41 sec	43 sec	38 sec	47 sec

Table 6.10: Summary of simulation results for Rule set 4, Case 1, 2, 3 and 4 with Seed 2.

As indicated in Table 6.6, Rule set 4 is calculated to have the lowest total cost for all cases except for Case 1. In Case 1, Rule set 4 has the highest number of days where the vessel is in operation. The reason is that the rule set book visits solely when the inventory levels go below the given limit of 600 tons. In the case study, the start inventory levels are set to 1700 ton, 1400 ton, and 1100 ton for the exposed, normal and sheltered location, respectively. Based on the inventory predictions, the rule set will first book a visit for the sheltered, then the normal, and lastly, the exposed. They will, in other words, be booked on separate days. Since the consumption rate is constant, all future visits will also be on separate days, and the option of two visits during one voyage will not be exploited. The other rule sets, which are based on the last visit to each location, will take advantage of the option of multiple visits from the beginning since the last visits are set to zero and therefore save charter costs. For Rule set 4, this is only apparent in Case 1. In other cases, the total cost is calculated to be lower compared to the other rule sets. The utilization of the vessel is still, lower compared to the other rule sets, with a 1.18 visits performed per day. Rule set 4 does in other words have a higher potential for improvement compared to the others, as already mentioned. If Rule set 4 could utilize the option of two visits on one voyage, by example, having different lower limits at each location or force the vessel to do two visits every time it sails, the cost is believed to be reduced.

Figure 6.8: Caption

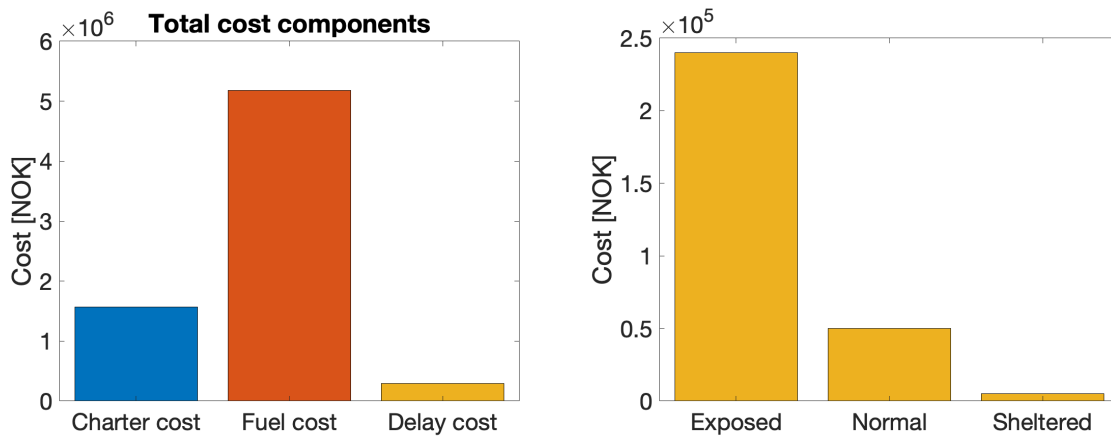


Figure 6.7: Total cost and delay cost for each of the locations for Rule set 4, Case 2, Seed 2.

As the uncertainty increases with the case number, the total cost with Rule set 4 first decrease from Case 1 to 2, but then increases. For Case 2 and 3, the delay cost is 100 000 NOK higher than for Rule set 3, as seen in Table 6.10. As can be seen in Figure 6.7, the delay cost is highest for the exposed location as predicted, but also the normal and sheltered cause some delay cost. Compared to the delay cost for Rule set 3, as seen in Figure 6.6, where only the exposed location cause delay costs, Rule set 4 also causes delays at the normal and the sheltered locations. Considering the fact that Rule set 4 includes the highest amount of information among the different rule sets, it would be predicted that Rule set 4 would be able to avoid delay costs in general, especially at the sheltered location. The reason for this is unknown.

Rule set 4 generates the highest delay cost, but also the lowest fuel cost and days in operation. These are all connected. Fewer visits planned, fewer days in operation and lower fuel costs. In the long run, a low number of days in operation might be ideal due to lower wear and tear on both the vessel and the facilities. The importance of no delay costs versus lower number of days in operation must therefore be evaluated. For a feed distribution problem, it would be critical for a location to run out of feed, so the long term impact should be evaluated.

Overall, Rule set 4 induce a lower total cost compared to the other rule sets. However, it also induces one of the higher delay costs. If the aim of the rule sets is to prevent delay costs, Rule set 4 is not optimal. If the aim is to find the most cost-efficient rule set, Rule set 4 is the best.

As demonstrated, a rule-based method can be used to generate plans for the routing of priority-based operations. The performance of the rule sets shows that Rule set 4 generates overall less costly plans for a one-year simulation when weather conditions are included. At the same time, Rule set 4 generate the highest delay cost. In order to assess the optimality of the solutions generated, the results should be compared to results with the use of another method.

Amount Transported per Day in Operation

Another simulation output that show some interesting results are the amount of feed transported. The histograms in Figure 6.9 show the distribution of total amount transported every day the vessel is in operation. As is illustrated there, Rule set 4 mostly transports 1800 tons per day, while the other rule sets mostly transports 1500 tons. This is due to the set lower limit of 600 tons for the inventory in Rule set 4. Rule set 4 is in other words more efficient with regards to amount transported compared to the others. But, all rule sets except for Rule set 4 transport up to 2700 tons more often than Rule set 4. This support the earlier claim that Rule set 4 has a higher potential for improvement.

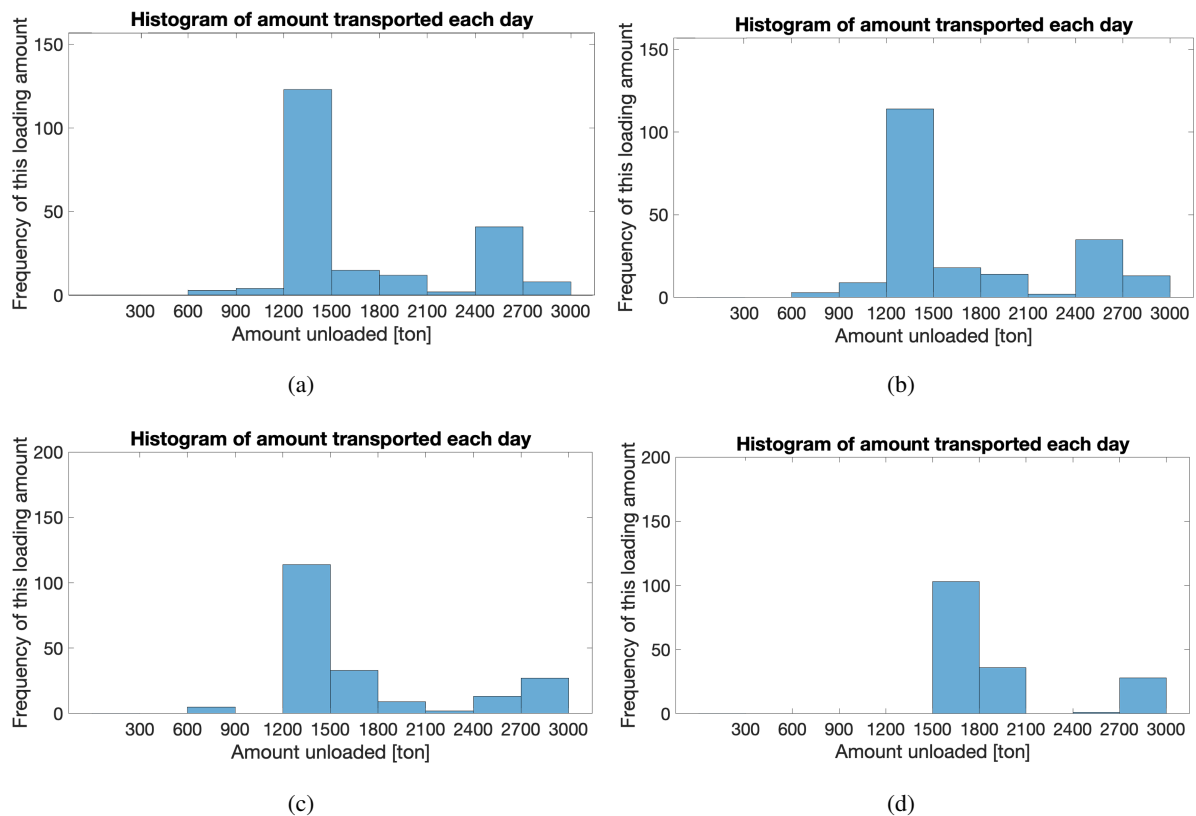


Figure 6.9: Histogram of the amount of feed transported each day in operation for (a) Rule set 1, (b) Rule set 2, (c) Rule set 3 and (d) Rule set 4.

6.5 Evaluation of Value of Information

The amount of information available increases with the rule sets. Rule set 1 has the lowest amount of information, while Rule set 4 has the highest amount. In order to look at the value of information, the total cost for each case for each rule set should be compared.

In Case 1, where the weather is perfect, there is no value in knowing the weather conditions. The total costs are equal for Rule set 1, 2, and 3, while Rule set 4 generates an increased cost, as seen in Table 6.6. This is due to the start-up levels, as mentioned earlier. The benefit of knowing the inventory levels is, therefore, unclear since the cost depends on the start-up levels.

Case 2 introduces a perfect weather forecast; it is, therefore, possible to plan based on the next two weeks for Rule set 3, without any uncertainty. The total costs in Case 2 are reduced from Rule set 1 to Rule set 4, with a small increase in cost for Rule set 2. This increase is due to a higher charter cost because of additional days in operation compared to Rule set 1. This is most probably due to the possibility of moving the next visit. It is, therefore, no value of knowing the weather forecast for the next day, but costs can be saved if the weather forecast for the next two weeks is known. The cost is reduced by around 400,000 NOK when the weather forecast for two weeks is known. When the inventory levels are known additionally, an extra 1,5 million NOK can be saved. The value of knowing the inventory levels are therefore high, compared to the weather forecast. However, since the weather forecast for the normal location is not as harsh as it should be, the effect of knowing the weather forecast may be larger if the weather forecast is corrected. Case 3 and 4 show the same development of the total cost as in Case 2. The cost generally decreases except for an increase with Rule set 2.

In conclusion, the total cost is decreased as more information is obtained. Information about the current inventory levels makes up the highest reduction cost with up to 1,900,000 NOK, while the information about the weather only saves at most 600,000 NOK. It is on the other hand believed that if the normal location is more unavailable, the value of information would become more important.

This extra information does not come for free, and measures must be taken to collect it. Metocean information can be obtained by setting up a weather buoy in the desired location, preferably as early as possible, to ensure that long time series with data can be available. The cost of setting up such a construction should be compared with the predicted savings of having this information. In the long run, such information is believed to be cost-efficient.

In order to know the inventory levels at each location, sensors that measure the amount of feed distributed and unloaded from the vessel should be mounted. Such sensors or systems that control the feed levels at aquaculture facilities are normally already integrated. This should also be included in other facilities that fit the problem description.

7 | Discussion

This chapter discusses the assumptions and decisions made during the work with the model and thesis in general. The discussion also assess the results from the case study and evaluate if they represent the model as intended.

7.1 Modelling Assumptions

During the model generation, with regards to both rule sets and the simulation model, some assumptions had to be made.

Problem Description

In the problem description, the classification of exposed, normal, and sheltered locations is presented. This classification is included in order to have a specific measure for when a location should be considered as exposed. Bad weather is defined as weather that exceeds the operational limit of the vessel. In this case, this classification is used to set the operational limit for the vessel based on the given data set. A sea state of 3 is chosen as the operational limit in Chapter 6 after observing when the likelihood of bad weather exceeds 20%. In other words, an exposed location may be unavailable 1 out of 5 days. This method of choosing the operational limit based on availability is considered to be logical. The normal method for choosing a vessel is based on the desired availability at the site. The set limit of when a location is considered as exposed may affect the results because this affects the availability. Since the limit stays constant in all the cases, it is assumed that it does not affect the possibility of comparing the results across cases and rule sets.

In the Chapter 2, the total cost of the routing is defined to be calculated based on charter, fuel, and delay costs. The charter cost is assumed to be per day the vessel is in use, the fuel cost per nautical mile traveled, and the delay cost per day the locations are without their needed inventory. In relation to the charter cost, it is assumed that the vessel can be chartered whenever needed, which might be unrealistic unless the vessel is positioned in the area and have very few other operations. For the kind of problem that is presented here, where the vessel may be utilized in other locations owned by the company chartering the vessel, it is likely that it is possible to use the vessel when needed. The fuel cost, which is set to only depend on the distance traveled, would, in reality, depend on the total fuel consumption. The fuel consumption of a vessel depends on the machinery of the vessel, machinery horsepower, and fuel type. The fuel cost practiced here is, therefore, a major simplification, but as the same calculation is practiced

in all Cases and for all Rule sets, the basis for comparison is still assumed to be valid. The delay cost is assumed to be equal for all locations and dependent on the days where the storage at the locations are empty. In retrospect, the delay cost should be practiced once the inventory level reaches the lower limit in order to avoid empty storage. As the delay cost is presented now, it represents the number of days the locations have empty storage.

The test scenarios performed in the Exploring the Problem was a nice way of learning more about the problem dynamics and the factors influencing the decision process. The relevance and transferability to a real-life situation can be questioned since the test and results are based on the authors' knowledge and logic reasoning. Since the case scenarios were only used as inspiration for which kind of factors that are important in the decision process, they are not considered to affect the results.

Rule Sets

The rule sets generated and used in the routing are based on the type of information available. They are made up of several IF- and FOR-loops inside the MATLAB environment, making it possible to control the rule set completely.

In rule set 1 it is assumed that the only information accessible is the consumption rate and the storage capacity of the vessel and each location. Such an assumption is not very realistic, the weather and inventory levels are often known, but due to the objective of the thesis, it is logical to include. The consumption rate is also assumed to stay constant through a one-year simulation, which might not be realistic if the inventory considered is, in example, feed. The fish would require an increased amount of feed as it grows, and the consumption rate should, therefore, change throughout the year. In general, the model would become more robust if uncertainty in the consumption rate is included. A constant consumption rate is not considered to generate wrong results, but an uncertain consumption rate might make a more robust model.

Rule set 2 has information about the weather forecast the following day. This forecast can be used to check if the location of where a visit is booked is available. If not, the visit is moved to the day after, unless a visit to a location with a higher priority is booked on this day, then the visit with the lowest priority is deleted. Since it is mostly the exposed location that is unavailable, visits are rarely deleted. In the other rule sets, a visit is deleted if the location is unavailable on the day of operation. In retrospect, such a rule should not have been included. All rule sets should be based on the same.

The plan generated by the rule sets is for the next 14 days. If the weather forecast has changed, and the plan must be updated, the new plan only updates the remaining part of the plan. In a real-life scenario, the plan would not only get updated but also extended. Since the weather forecast is known for the next 14 days for Rule set 3 and 4, it would be natural to generate a new 14-day long plan, independently of where in the current plan the rerouting happens.

If the generation of a plan also constitutes a cost, the model should be changed so that the new plan is not generated unless changed weather conditions for the upcoming days. The likelihood of change in the weather forecast 13 days into the future is big, and a new plan should, therefore, not be generated only based on this if costs are induced.

In order to know the value of knowing only the inventory levels, a new rule set should be made. This

should base the routing solely on the current inventory levels.

Weather Representation

The weather is generated based on the principle of Markov chains and transition probabilities. As mentioned, the transition matrix is based on 10 years of weather data. The weather data consist of the significant wave height from wind and swell. Wave direction and wind are not included in the analysis since the wave height is considered to be the dominant factor for offshore operations.

In order to differentiate the weather experienced at the exposed and the normal location, the weather matrix for the normal location was scaled based on the exposed matrix. The weather generated for the normal location is reduced with two sea states lower than the exposed. In retrospect, it should have decreased with only one sea state lower in order to get a more varied weather condition. Scaling the weather at the normal location has affected the results since this change the availability at the normal location. As illustrated in Section 6.3, the normal location is almost as available as the sheltered location. The weather at the normal location should be more uncertain to get a more robust model, and scaling should, therefore, be avoided. In order to make the simulation more realistic, a new transition matrix based on weather data for a similar location should be used as a basis for the normal location.

The sheltered location was assumed to always be available. The correctness of this assumption depends on how sheltered the location is. In order for the full availability to be correct, it must never experience harsh weather, which might be the case for facilities in the fjords.

The weather forecast is generated based on the real weather for each location. In order to introduce uncertainty to the model, the forecast is changed with random values included in the model. Values based on the normal distribution are added or subtracted to the model. The uncertainty of the forecast is made higher the further into the future. The use of random values for the weather forecast introduces uncertainty, but since the random values used to change the forecast are not linked, the resulting weather forecast may be without correlation. This does not reduce the correctness of the forecast since only one value per day is considered to represent a full day. In order to have a more robust model, the weather forecast for the whole day should be included. For the objective of the model, assessing the method generated, and the value of information, information regarding the weather for the full day may result in an even better routing of the vessel.

As mentioned, Markov chains were chosen over the probability distribution in the method for weather generation. The main reason for this choice was the desired correlation between the values. Since the routing ended up being based on only one value per day, the correlation is slightly less meaningful. However, the correlation between the consecutive days is still necessary in order to have realistic weather, and the choice therefore, still stands.

7.2 Simulation Model

The simulation model is used to test the robustness of the method developed and the value of information. As mentioned, the model is made inside Simulink as a discrete event simulation model. The model is driven by events, not by time. The upper loop of the model, where the operations are conducted, is

not dependent on time. The time it takes to complete the operations are all set to zero. If no operation is booked, the vessel waits 24 hours before the next day in the plan is checked. The decision of not including time was based on the fact that the purpose of the simulation was to test how good the method for routing was. In order to do so, time does not have to be included. Excluding time from the simulation have some implications for the rest of the model. First of all, the sailing time is only calculated based on the distance traveled and the speed of the vessel. The speed would, in reality, be affected by the weather conditions and the time of the voyage would therefore also vary. Since time is not included, this is not considered.

Another complication that arises when time is not included is the question of whether the vessel manages to complete the operations during the time available. If two visits are booked on the same voyage, this model does not calculate the time of each operation; it is only assumed to be enough time. In order to make the model more robust, the time of each operation should be included in the planning process. The same applies to the loading and unloading time at each location. The time of unloading depends on the amount unloaded. Since time is not included, this is not accounted for.

As previously mentioned, the first plan for the vessel is generated in MATLAB outside Simulink and then imported to Simulink. In order to simplify the model, all of the plans should be generated inside Simulink. It is not believed to affect the results of the model, besides maybe the run time.

The function inside Simulink generates a new plan every time the availability is changed. This should have been different for the various rule sets. For Rule set 1, a new plan should only be generated at the end of the planning period. For Rule set 2, a new plan should be generated only if the availability the next day changes. The code used compares the availability for the next to weeks for all rule sets. This is why the number of new plans generated is the same for all rule sets. It was predicted that number of plans generated would be dependent on the rule set and the information given, and this, therefore, explains why they did not change.

The uncertainty included in this simulation model is only the weather conditions. The model would be more robust if it includes uncertainty in the consumption at each location, available feed at the port, or unexpected incidents such as engine failure.

Three different seed values were used in the simulation to create a larger basis for comparison. The results showed that the costs vary a lot from one seed to another. The average costs were calculated based on the various values, but since the trend was that the cost increased, this average is not representative of all simulations. In order to get a better overview of the trends related to various seeds, and the overall simulation results independently from seed, the simulation should be run with several various seeds. In Tables 6.7, 6.8, 6.9 and 6.10, the simulation results with Seed 2 is presented as an example, just because the values with Seed 2 were normally close to the average calculated.

As mentioned in Section 5.6, verification of the model was made while modeling. In order to validate that the model represents the real problem, it should be compared with a real-life problem.

7.3 Simulation Results

The results presented in Section 6.4 show that information about the inventory level saves about 2 million NOK in costs, this is as predicted. The information about the weather does not have the same effect, which is not as predicted. The costs are strongly related to the rates chosen for charter, fuel and delay costs. The relationship between these is, therefore, believed to affect the results.

As discussed in 6.4, Rule set 4 has a high potential for improvement, which also support the argument of high value in the information about the inventory levels. If the vessel are utilized both with regards to amount transported and days in operation, the rule set may be able to find optimal results.

The results still present the differences in the information given, which was the intention. The cases which were used to test the rule sets should have been even more distinctive in order to distinguish the rule sets ability to generate good plans.

8 | Conclusion

The objective of this thesis was to gain insight into the priority-based routing problem and investigate possible routing methods. The value of information in the routing process should be evaluated as a byproduct to this. The intention behind the thesis is to investigate a less studied area, in order to pave the way for future studies.

The rule-based method was chosen as the preferred routing method after also assessing exact and approximated methods. In order to assess the value of information, four rule sets with access to different amounts of information are made. The rule sets are based on rules explored and found through several manual cases, including priority, weather conditions, and inventory levels. In order to test the performance of each rule set, a simulation model is generated in Simulink. The simulation model includes the influence of weather on availability. The rule sets are tested with four degrees of uncertain weather conditions. In order to assess various realizations of the weather, three seeds are used in the simulations.

A case study involving the feed distribution to aquaculture facilities outside the coast of Norway is conducted to test the rule sets. The results demonstrated that a rule-based method for routing is possible. The performance of the rule sets proved that Rule set 4, which take both priority, weather and inventory into consideration, generates the most cost-efficient plans, but also induce the highest delay cost. It was also discovered that there is a high potential for improvement in Rule set 4 with regards to utilization of the vessel. This supports the hypothesis that more information leads to more efficient plans, although the information about the weather did not reduce the costs as much as predicted. The various cases proved that uncertainty in weather forecast does not induce as large differences in cost as predicted.

The work performed in this master thesis has increased the knowledge of both the priority-based routing problem and the use of a rule-based method in routing. The findings may be used in further analysis of the problem, and maybe in a future decision support model for industries with priority-based operations. Such a tool will become important as industries are pursuing opportunities further offshore.

8.1 Further Work

During the work with the thesis, some limitations are made due to limited information and time. This problem has, as mentioned, received little attention so far, and a big part of the process has therefore been to define the problem. The limitations of the problem are set by the author in order to cover the most important points, such as the critical routing factors and the methods for routing. The value of information is assessed implicitly through the testing of the rule sets.

The rule sets should be further improved by assessing the information separately in addition to combine. It would, for example, be interesting to look at the results from a rule set only looking at the inventory levels.

A sensitivity analysis for the parameters used in the case study should be conducted to assess the degree of influence of each value. As mentioned, the start values of the inventory levels lead to a specific result. The cost rates should also be evaluated. A sensitivity analysis would provide a better foundation for comparison of the rule sets.

If the model includes uncertainty in other areas than the weather, the robustness of the methods can be further tested. Suggestions to where uncertainty can be included are the feed consumption, feed availability at port, or problems with the vessel. By including more uncertainty, a more robust method may be found.

Lastly, the model should be modified so that a new plan is not generated unless the weather requires so.

Bibliography

- Agra, A., Christiansen, M., Hvattum, L.M., Rodrigues, F., 2018. Robust Optimization for a Maritime Inventory Routing Problem. *Transportation Science* 52, 509–525.
- Agra, A., Christiansen, M., Ivarsøy, K.S., Solhaug, I.E., Tomaszgard, A., 2017. Combined ship routing and inventory management in the salmon farming industry. *Annals of Operations Research* 253, 799–823. URL: <https://doi.org/10.1007/s10479-015-2088-x>, doi:10.1007/s10479-015-2088-x.
- Avriel, M., Williams, A.C., 1970. The Value of Information and Stochastic Programming. *Operations Research* 18, 947–954. URL: www.jstor.org/stable/169369. publisher: INFORMS.
- Borthen, T., Loennechen, H., Wang, X., Fagerholt, K., Vidal, T., 2018. A genetic search-based heuristic for a fleet size and periodic routing problem with application to offshore supply planning. *EURO Journal on Transportation and Logistics* 7, 121–150. URL: <https://doi.org/10.1007/s13676-017-0111-x>, doi:10.1007/s13676-017-0111-x.
- College, D., . Chapter 11 Markov Chains, Dartmouth College. URL: https://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/Chapter11.pdf.
- Copernicus, 2019. ERA5 hourly data on single levels from 1979 to present. URL: <https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=form>.
- Fagerholt, K., Hvattum, L., Johnsen, T., Korsvik, J., 2013. Routing and scheduling in project shipping. *Annals of Operations Research* 207, 67–81. Place: Boston Publisher: Springer US.
- Fagerholt, K., Lindstad, H., 2007. TurboRouter: An Interactive Optimisation-Based Decision Support System for Ship Routing and Scheduling. *Maritime Economics & Logistics* 9, 214. Publisher: Palgrave Macmillan.
- Google, 2020. Frohavet - Google Maps. URL: <https://www.google.com/maps/place/63%C2%B055'48.0%22N+9%C2%B004'48.0%22E/@63.9300023,9.0778113,17z/data=!3m1!4b1!4m5!3m4!1s0x0:0x0!8m2!3d63.93!4d9.08>.

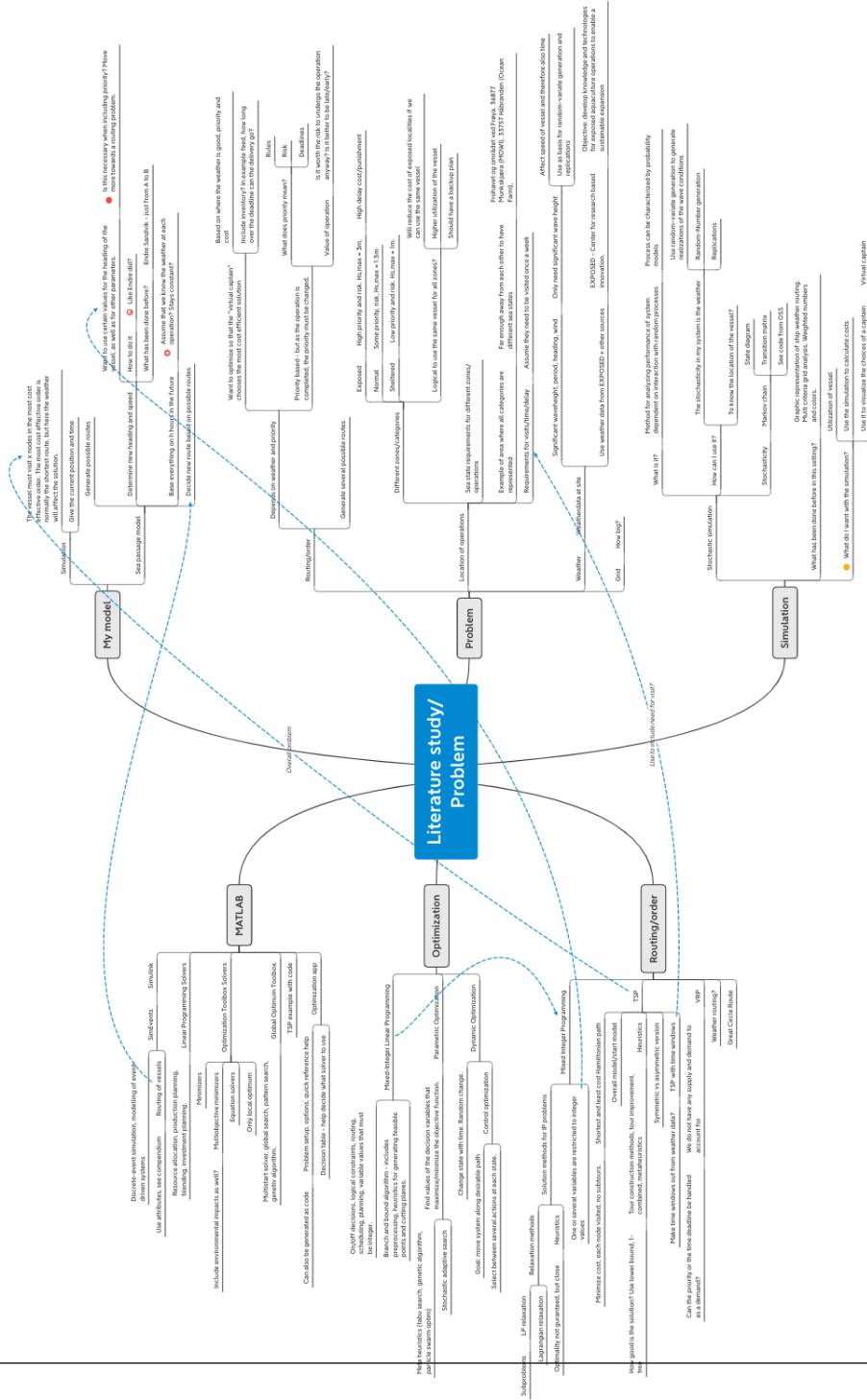
-
- Grifoll, M., Martínez de Osés, F., Castells, M., 2018. Potential economic benefits of using a weather ship routing system at Short Sea Shipping. *WMU Journal of Maritime Affairs* 17, 195–211. Place: Berlin/Heidelberg Publisher: Springer Berlin Heidelberg.
- Halvorsen-Weare, E.E., Fagerholt, K., 2011. Robust Supply Vessel Planning, in: Pahl, J., Reiners, T., Voß, S. (Eds.), *Network Optimization*, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 559–573.
- Halvorsen-Weare, E.E., Fagerholt, K., Nonås, L.M., Asbjørnslett, B.E., 2012. Optimal fleet composition and periodic routing of offshore supply vessels. *European Journal of Operational Research* 223, 508 – 517. URL: <http://www.sciencedirect.com/science/article/pii/S0377221712004729>, doi:<https://doi.org/10.1016/j.ejor.2012.06.017>.
- Haupt, R., 1989. A survey of priority rule-based scheduling. *Operations-Research-Spektrum* 11, 3–16. Place: Berlin/Heidelberg Publisher: Springer-Verlag.
- Hillier, F.S., Lieberman, G.J., 2015. *Introduction to operations research*. 10th ed. ed., McGraw-Hill, New York.
- Kisialiou, Y., Gribkovskaia, I., Laporte, G., 2018. Robust supply vessel routing and scheduling. *Transportation Research Part C* 90, 366–378. Publisher: Elsevier Ltd.
- Kjølleberg, J., 2015. *Weather Routing of Supply Vessels in the North Sea - Solving the Supply Vessel Weather Routing Problem*. Ph.D. thesis. NTNU. URL: <http://hdl.handle.net/11250/2350788>.
- Kystverket, 2020. *Kystinfo - Akvakulturanlett med innehaver*. URL: <https://a3.kystverket.no/kystinfo>.
- MathWorks, a. *SimEvents*. URL: <https://se.mathworks.com/products/simevents.html>. library Catalog: se.mathworks.com.
- MathWorks, b. *Simulink Documentation - MathWorks Nordic*. URL: <https://se.mathworks.com/help/simulink/>.
- Pisinger, D., Ropke, S., 2010. Large Neighborhood Search, in: *Handbook of Metaheuristics*, pp. 399–419. doi:10.1007/978-1-4419-1665-5_13.
- Sandvik, E., Nielsen, J.B., Asbjørnslett, B.E., Pedersen, E., Fagerholt, K., 2019. Operational sea passage scenario generation for virtual testing of ships using an optimization for simulation approach. Manuscript submitted for review. .
- Szlapczynski, R., 2006. A New Method of Ship Routing on Raster Grids, with Turn Penalties and Collision Avoidance. *The Journal of Navigation* 59, 27–42. URL: <https://search.proquest.com/docview/229666153?accountid=12870>. place: Cambridge Publisher: Cambridge University Press.

UN, 2019. Growing at a slower pace, world population is expected to reach 9.7 billion in 2050 and could peak at nearly 11 billion around 2100. URL: <https://www.un.org/development/desa/en/news/population/world-population-prospects-2019.html>.

Wilson, E.C.F., 2015. A Practical Guide to Value of Information Analysis. *PharmacoEconomics* 33, 105–121. URL: <https://doi.org/10.1007/s40273-014-0219-x>, doi:10.1007/s40273-014-0219-x.

Appendix

A | Mindmap



B | Case A

Case A

What would I chose?

Do not consider weather the next week, only this day.

Set up different scenarios with different inventory level, artificial weather.

Systematic approach.

Do not consider the time of each travel here.

Do not include/consider long delays.

Priority is held constant due to problem definition.

Scenario	A - EXPOSED			B - NORMAL			C - SHELTERED			My choice:	Reason
	Inventory level	Weather conditions	Priority	Inventory level	Weather conditions	Priority	Inventory level	Weather conditions	Priority		
1	High	Good	High	High	Good	Medium	High	Good	Low	-	Stay in port
2	Medium	Good	High	High	Good	Medium	High	Good	Low	A	Inventory + priority.
3	Low	Good	High	High	Good	Medium	High	Good	Low	A	Inventory.
4	High	Good	High	Medium	Good	Medium	High	Good	Low	B	Inventory.
5	High	Good	High	Low	Good	Medium	High	Good	Low	B	Inventory.
6	High	Good	High	High	Good	Medium	Medium	Good	Low	C	Inventory.
7	High	Good	High	High	Good	Medium	Low	Good	Low	C	Inventory.
9	High	Medium	High	High	Good	Medium	High	Good	Low	A	Inventory. Likelihood for long period of bad weather, should fill up inventory before that period.
10	High	Bad	High	High	Good	Medium	High	Good	Low	Stay in port.	No need for visits.
11	High	Good	High	High	Medium	Medium	High	Good	Low	B	Inventory. Likelihood for long period of bad weather, should fill up inventory before that period.
12	High	Good	High	High	Bad	Medium	High	Good	Low	Stay in port.	No need for visits.
13	High	Good	High	High	Good	Medium	High	Medium	Low	Stay in port.	No need for visits.
14	High	Good	High	High	Good	Medium	High	Bad	Low	Stay in port.	No need for visits.
15	Medium	Good	High	Medium	Good	Medium	Medium	Good	Low	A	Inventory + priority.
16	High	Good	High	Medium	Good	Medium	Medium	Good	Low	B	Inventory + priority.
17	Medium	Good	High	Medium	Good	Medium	High	Good	Low	A	Inventory + priority.
18	Medium	Good	High	High	Good	Medium	Medium	Good	Low	A	Inventory + priority.
19	Low	Good	High	Low	Good	Medium	Low	Good	Low	A	Priority.
20	Low	Good	High	Low	Good	Medium	High	Good	Low	A	Priority.
21	Low	Good	High	High	Good	Medium	Low	Good	Low	A	Priority.
22	High	Good	High	Low	Good	Medium	Low	Good	Low	B	Priority.
23	Medium	Good	High	High	Good	Medium	Low	Good	Low	C	Inventory.
24	Medium	Good	High	Low	Good	Medium	High	Good	Low	B	Inventory.
25	Medium	Medium	High	Medium	Medium	Medium	Medium	Medium	Low	A	Priority and inventory.

Figure B.1: Case A.

C | Case B

Case B

Weather from KI 06:00 from nc-data
Consider weather known the next week.

Plan for one week ahead.

Results logged in the bottom can be placed into solution to see change in level.

Use same weather for all locations. Use operational limit to differentiate the availability.

Start at chosen inventory levels.

Only know the weather forecast for the next week.

Can only go to one location each day.

The columns are programmed to update based on choice of location

	Exposed	Normal	Sheltered
Priority	High	Medium	Low
Operatio	2	3	Always
Inventory	2000	2000	2000
Inventory	300	300	300

Reasons to choose from:

Inventory

Priority

Weather

No place to go

Combinations of these.

Week	Day	Hs [m]	A - Exposed		B - Normal		C - Sheltered		Next stop:	Reason:
			Inventory level	Availability	Inventory level	Availability	Inventory level	Availability		
1	Monday	1,18	1700	1	1700	1	1700	1	B	Inventory + Weather
1	Tuesday	1,86	1400	1	2000	1	1400	1	A	Inventory + Weather
1	Wednesday	2,72	2000	0	1700	1	1100	1	C	Inventory
1	Thursday	4,09	1700	0	1400	0	2000	1	-	No place to go
1	Friday	2,61	1400	0	1100	1	1700	1	B	Inventory
1	Saturday	2,91	1100	0	2000	1	1400	1	C	Inventory
1	Sunday	3,5	800	0	1700	0	2000	1	-	No place to go
2	Monday	1,45	500	1	1400	1	1700	1	A	Inventory
2	Tuesday	1,47	2000	1	1100	1	1400	1	B	Inventory
2	Wednesday	1,03	1700	1	2000	1	1100	1	C	Inventory
2	Thursday	1,8	1400	1	1700	1	2000	1	A	Inventory
2	Friday	1,73	2000	1	1400	1	1700	1	B	Inventory
2	Saturday	1,83	1700	1	2000	1	1400	1	A	Inventory + weather
2	Sunday	6,2	2000	0	1700	0	1100	1	C	Inventory
3	Monday	1,9	1700	1	1400	1	2000	1	B	Inventory + weather
3	Tuesday	1,62	1400	1	2000	1	1700	1	A	Inventory
3	Wednesday	3,55	2000	0	1700	0	1400	1	C	Inventory
3	Thursday	2,28	1700	0	1400	1	2000	1	B	Inventory
3	Friday	4,06	1400	0	2000	0	1700	1	C	Inventory + weather
3	Saturday	2,96	1100	0	1700	1	2000	1	B	Inventory + weather
3	Sunday	3,25	800	0	2000	0	1700	1	C	Inventory + weather
4	Monday	3,84	500	0	1700	0	2000	1	-	No place to go
4	Tuesday	2,24	200	0	1400	1	1700	1	B	Inventory
4	Wednesday	1,54	-100	1	2000	1	1400	1	A	Inventory
4	Thursday	1,3	2000	1	1700	1	1100	1	C	Inventory
4	Friday	0,93	1700	1	1400	1	2000	1	B	Inventory
4	Saturday	2,03	1400	0	2000	1	1700	1	C	Inventory
4	Sunday	1,47	1100	1	1700	1	2000	1	A	Inventory
5	Monday	1,99	2000	1	1400	1	1700	1	B	Inventory
5	Tuesday	2,11	1700	0	2000	1	1400	1	C	Inventory
5	Wednesday	3,06	1400	0	1700	0	2000	1	-	No place to go
5	Thursday	4,47	1100	0	1400	0	1700	1	C	Inventory + weather
5	Friday	2,17	800	0	1100	1	2000	1	B	Inventory
5	Saturday	4,1	500	0	2000	0	1700	1	C	Inventory + weather
5	Sunday	5,73	200	0	1700	0	2000	1	-	No place to go

Figure C.1: Case B

LOG

Run	1			2			3			4			5			
Comment																
Start level A Exposed	1100			1100			500			1700			1700			
Start level B Normal	1100			1100			200			1700			1700			
Start level C Sheltered	1100			1100			800			1700			1700			
Day\Hs																
Day	Hs [m]	Next stop	Reason	Extra comment	Next stop	Reason	Extra comment	Next stop	Reason	Extra comment	Next stop	Reason	Extra comment	Next stop	Reason	Extra comment
							Wait with A									
Monday	1,18	B	Inventory + weather		B	Inventory due to + Weather	weather	B	Inventory	-	No place to go			-	No place to go	Due to limit of 900
Tuesday	1,86	A	Inventory + weather		A	Inventory + Weather		A	Inventory	A	Weather			-	No place to go	
Wednesday	2,72	C	Inventory		C	Inventory		C	Inventory	B	Inventory + Priority	B and C equal inv lev, pri dominant.		B	Inventory + Priority	
Thursday	4,09	-	No place to go		-	No place to go		-	No place to go	C	Inventory			C	Inventory	
Friday	2,61	B	Inventory		B	Inventory		B	Inventory	-	No place to go			-	No place to go	
Saturday	2,91	C	Inventory		-	No place to go		-	No place to go	B	Weather			-	No place to go	
Sunday	3,5	-	No place to go		C	Inventory		C	Inventory	-	No place to go			-	No place to go	
Monday	1,45	A	Inventory		A	Inventory		A	Inventory	A	Inventory			A	Inventory	
Tuesday	1,47	B	Inventory		B	Inventory		B	Inventory	C	Inventory			B	Inventory	
Wednesday	1,03	C	Inventory		-	No place to go		-	No place to go	B	Inventory			C	Inventory	
Thursday	1,8	A	Inventory		C	Inventory		C	Inventory	-	No place to go			-	No place to go	
Friday	1,73	B	Inventory		A	Inventory		-	No place to go		Bad weather is coming.			C	Weather	Deliver before bad weather
Saturday	1,83	A	Inventory + weather		B	Inventory		A	Weather	A	Inventory + Weather			A	Weather + Priority	
Sunday	6,2	C	Inventory		-	No place to go		-	No place to go	-	No place to go			C	Inventory	

Figure C.2: Case B log.

D | Case C

Case C

Want to include time and cost of delay/fuel/etc

What is the strategy if the storm moves?

What should the relationship between the costs be?

Generate different scenarios with regards to weather, inventory level, delay cost vs fuel cost and time

Look at the "specific" scenarios, where tough choices have to be made based on cost and efficiency.

	A - Exposed	B - Normal	C - Sheltered
Operational limits	2	3	Always
Operational time	1	1	1

Availability:
Can go to site
Close to no go
No go

Fuel cost	1000
Charter cost	
Vessel capacity	60% of inventory
Length day	12
Inventory usage [./day]	10%

* If NOK/h is used, V= constant

Sailing time	Port	A	B	C
Port	0	2,5	1,5	0,5
A	2,5	0	1	2
B	1,5	1	0	1
C	0,5	2	1	0

BF = back and forth

Figure D.1: Case C details.

Description of scenario / day	A - Exposed				B - Normal				C - Sheltered				Comment:										
	Availability	Inventory level	Delay cost [\$/day]	Relations ship cost/Fuel cost (BF)	Availability	Inventory level	Delay cost	Relations ship cost/Fuel cost (BF)	Availability	Inventory level	Delay cost	Relations ship cost/Fuel cost (BF)		Total time [h]									
1 Low inventory	1	20%	20000	5000	4	5	6	1	20%	15000	3000	5	3	4	1	20%	10000	2000	5	2	3	A - B	Only time to do two. Choose A then B due to the delay cost/priority, C next day.
2 Low inventory and change of weather	1	20%	20000	5000	4	5	6	2	20%	15000	3000	5	3	4	1	20%	10000	2000	5	2	3	B - A	B closing in on bad weather? Do A afterwards, C next day.
3 On the way to B, bad weather occurs	1	40%	20000	5000	4	5	6	0	20%	15000	3000	5	3	4	1	40%	10000	2000	5	2	3	A - (C)	How far has the ship come. What does it have time for? Prioritize A.
4 Low delay cost/fuel cost relationship	1	10%	10000	5000	2	5	6	1	10%	9000	3000	3	3	4	1	10%	4000	2000	2	2	3	A - B	Can go to all sites. More expensive to have a delay at A than visiting B and C instead. Only "save" 2000 by visiting C compared to 5000 and 6000 at A and B.
5 Lower delay cost	1	10%	5000	5000	1	5	6	1	10%	5000	3000	1.7	3	4	1	10%	5000	2000	2.5	2	3	B - C	Same delay cost for all locations. Should visit B and C because that saves most cost. A has equal delay cost and travel cost.
6 Almost full inventory level	2	80%	7500	5000	1.5	5	6	2	80%	4500	3000	1.5	3	4	1	80%	3000	2000	1.5	2	3	Stay in port	Almost full inventory level. Closing in on some bad weather. Cost relationship 1.5. Should fill up A if bad weather is known to come. B and C does not need extra, too expensive compared to how much we get out.
7 Wait for weather vs go back																							Bad weather A, but it is supposed to lighten up. Should I travel there and then wait or do the two other locations instead? Cost 1000 /hour of waiting with the vessel. Is there a better option to do B and C instead. And then do A tomorrow.
8 Wait for weather vs go back, low inventory																							Delay cost at A. If we do not go there, it will be smart as long as we do not have to wait more than 2 hours. Wait here and not in scenario 7 because the inventory level is lower!

Immediate thoughts:
The delay cost mostly decide where the ship should go. If changes happen, the captain will go where the "value" is highest

Figure D.2: Case C log.

E | Rule sets

E.1 *Ruleset111*

```
%% RULE SET 11 - NO INFORMATION

% Only the consumption rate and storage capacities are known.
% Priorities are known.
% No weather conditions or inventory level are known.
% Book based on last visit.
% Allow for double visits.

% Preallocate values the first time
ExpLastVis = 0;
NormLastVis = 0;
ShellLastVis = 0;

% Can only change future visits, therefore count + 1
for i = count + 1:planningPeriod
    plan(i,1) = Day;

    % Set rest of columns to zero. Can be changed.
    plan(i,3) = 0;
    plan(i,4) = 0;

    % Book exposed location first due to priority
    % EXPOSED
    if ExpLastVis == 0           % Visit must be booked now.
        plan(i,2) = 1;
        ExpLastVis = i;
    elseif abs(i - ExpLastVis) >= 4
        plan(i,2) = 1;
        ExpLastVis = i;
    end

    % NORMAL
    if NormLastVis == 0         % Visit booked now
        if plan(i,2) == 0
```

```

        plan(i,2) = 2;
        NormLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 2;
        NormLastVis = i;
    end
elseif abs(i - NormLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 2;
        NormLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 2;
        NormLastVis = i;
    end
end
end

% SHELTERED
if ShellLastVis == 0
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
elseif abs(i - ShellLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
end
end

% Update the Day value to have correct value in column 1
Day = Day + 1;
end

```

E.2 *Ruleset112*

```

%% RULE SET 11 - NO INFORMATION

% Only the consumption rate and storage capacities are known.
% Priorities are known.
% No weather conditions or inventory level are known.
% Book based on last visit.
% Allow for double visits.

```

```

% Preallocate values the first time
ExpLastVis = 0;
NormLastVis = 0;
ShellLastVis = 0;

% Can only change future visits, therefore count + 1
for i = count + 1:planningPeriod
    plan(i,1) = Day;

    % Set rest of columns to zero. Can be changed.
    plan(i,3) = 0;
    plan(i,4) = 0;

    % Book exposed location first due to priority
    % EXPOSED
    if ExpLastVis == 0           % Visit must be booked now.
        plan(i,2) = 1;
        ExpLastVis = i;
    elseif abs(i - ExpLastVis) >= 4
        plan(i,2) = 1;
        ExpLastVis = i;
    end

    % NORMAL
    if NormLastVis == 0         % Visit booked now
        if plan(i,2) == 0
            plan(i,2) = 2;
            NormLastVis = i;
        elseif plan(i,3) == 0
            plan(i,3) = 2;
            NormLastVis = i;
        end
    elseif abs(i - NormLastVis) >= 4
        if plan(i,2) == 0
            plan(i,2) = 2;
            NormLastVis = i;
        elseif plan(i,3) == 0
            plan(i,3) = 2;
            NormLastVis = i;
        end
    end
end

% SHELTERED
if ShellLastVis == 0
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0

```

```

        plan(i,3) = 3;
        ShellLastVis = i;
    end
elseif abs(i - ShellLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
end
end

% Update the Day value to have correct value in column 1
Day = Day + 1;
end

```

E.3 *Ruleset113*

```

%% RULE SET 11 - NO INFORMATION

% Only the consumption rate and storage capacities are known.
% Priorities are known.
% No weather conditions or inventory level are known.
% Book based on last visit.
% Allow for double visits.

% Preallocate values the first time
ExpLastVis = 0;
NormLastVis = 0;
ShellLastVis = 0;

% Can only change future visits, therefore count + 1
for i = count + 1:planningPeriod
    plan(i,1) = Day;

    % Set rest of columns to zero. Can be changed.
    plan(i,3) = 0;
    plan(i,4) = 0;

    % Book exposed location first due to priority
    % EXPOSED
    if ExpLastVis == 0          % Visit must be booked now.
        plan(i,2) = 1;
        ExpLastVis = i;
    elseif abs(i - ExpLastVis) >= 4
        plan(i,2) = 1;
        ExpLastVis = i;
    end
end

```

```

end

% NORMAL
if NormLastVis == 0           % Visit booked now
    if plan(i,2) == 0
        plan(i,2) = 2;
        NormLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 2;
        NormLastVis = i;
    end
elseif abs(i - NormLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 2;
        NormLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 2;
        NormLastVis = i;
    end
end
end

% SHELTERED
if ShellLastVis == 0
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
elseif abs(i - ShellLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
end
end

% Update the Day value to have correct value in column 1
Day = Day + 1;
end

```

E.4 Ruleset114

```
%% RULE SET 11 - NO INFORMATION
```

```

% Only the consumption rate and storage capacities are known.
% Priorities are known.
% No weather conditions or inventory level are known.
% Book based on last visit.
% Allow for double visits.

% Preallocate values the first time
ExpLastVis = 0;
NormLastVis = 0;
ShelLastVis = 0;

% Can only change future visits, therefore count + 1
for i = count + 1:planningPeriod
    plan(i,1) = Day;

    % Set rest of columns to zero. Can be changed.
    plan(i,3) = 0;
    plan(i,4) = 0;

    % Book exposed location first due to priority
    % EXPOSED
    if ExpLastVis == 0           % Visit must be booked now.
        plan(i,2) = 1;
        ExpLastVis = i;
    elseif abs(i - ExpLastVis) >= 4
        plan(i,2) = 1;
        ExpLastVis = i;
    end

    % NORMAL
    if NormLastVis == 0         % Visit booked now
        if plan(i,2) == 0
            plan(i,2) = 2;
            NormLastVis = i;
        elseif plan(i,3) == 0
            plan(i,3) = 2;
            NormLastVis = i;
        end
    elseif abs(i - NormLastVis) >= 4
        if plan(i,2) == 0
            plan(i,2) = 2;
            NormLastVis = i;
        elseif plan(i,3) == 0
            plan(i,3) = 2;
            NormLastVis = i;
        end
    end
end

% SHELTERED

```

```
if ShellLastVis == 0
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
elseif abs(i - ShellLastVis) >= 4
    if plan(i,2) == 0
        plan(i,2) = 3;
        ShellLastVis = i;
    elseif plan(i,3) == 0
        plan(i,3) = 3;
        ShellLastVis = i;
    end
end

% Update the Day value to have correct value in column 1
Day = Day + 1;
end
```


F | Simulation Model

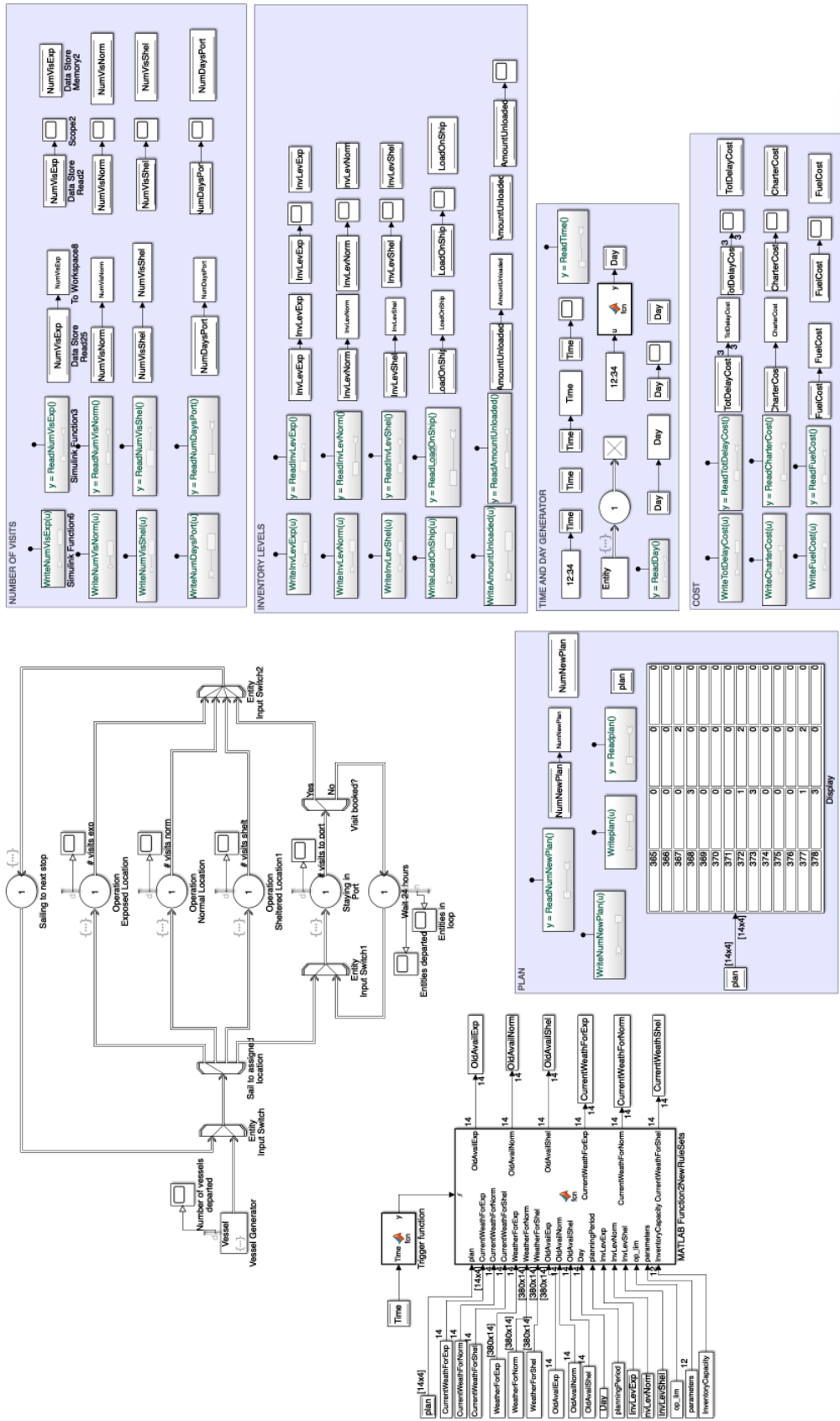


Figure F.1: Snapshot of the complete model in Simulink.

F.1 Routing Model up close

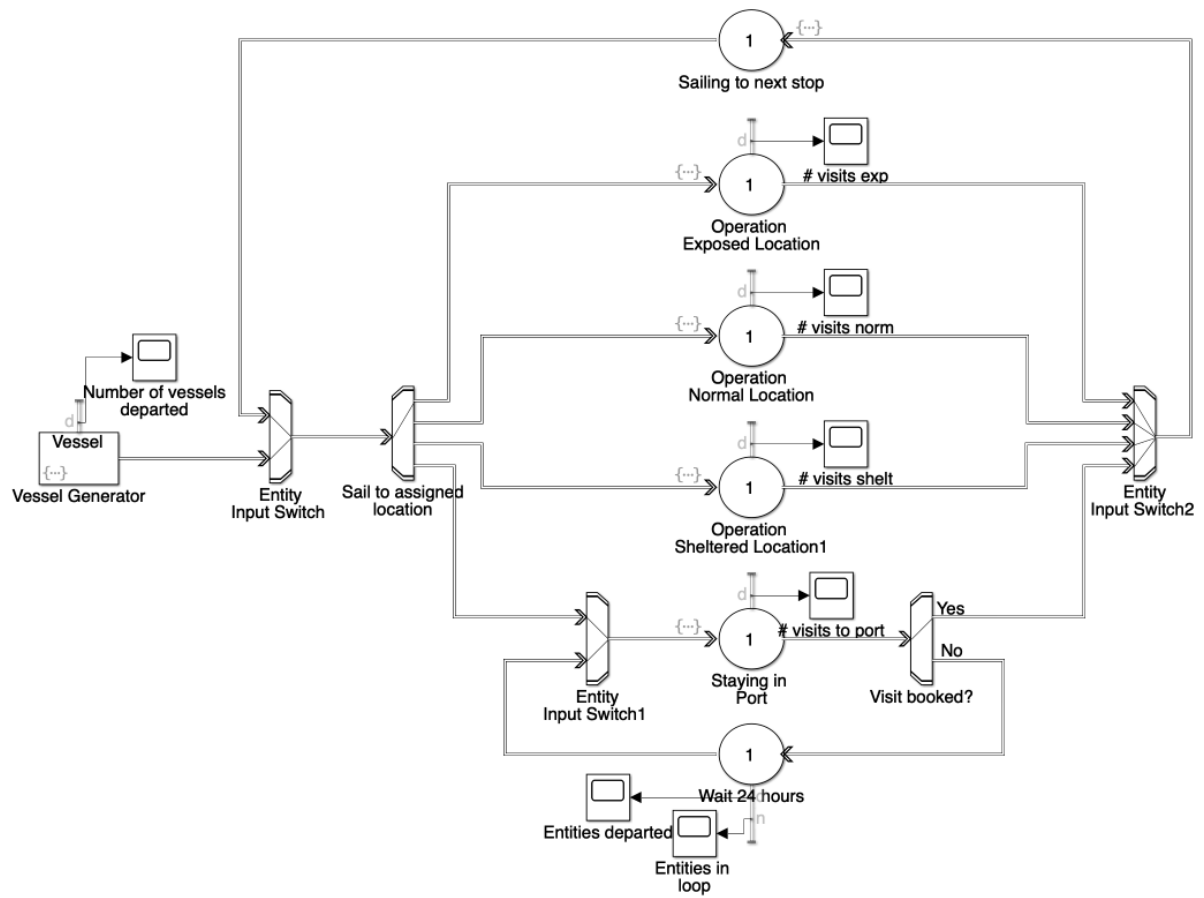


Figure F.2: Simulation model, snapshot from Simulink.

F.2 MATLAB function

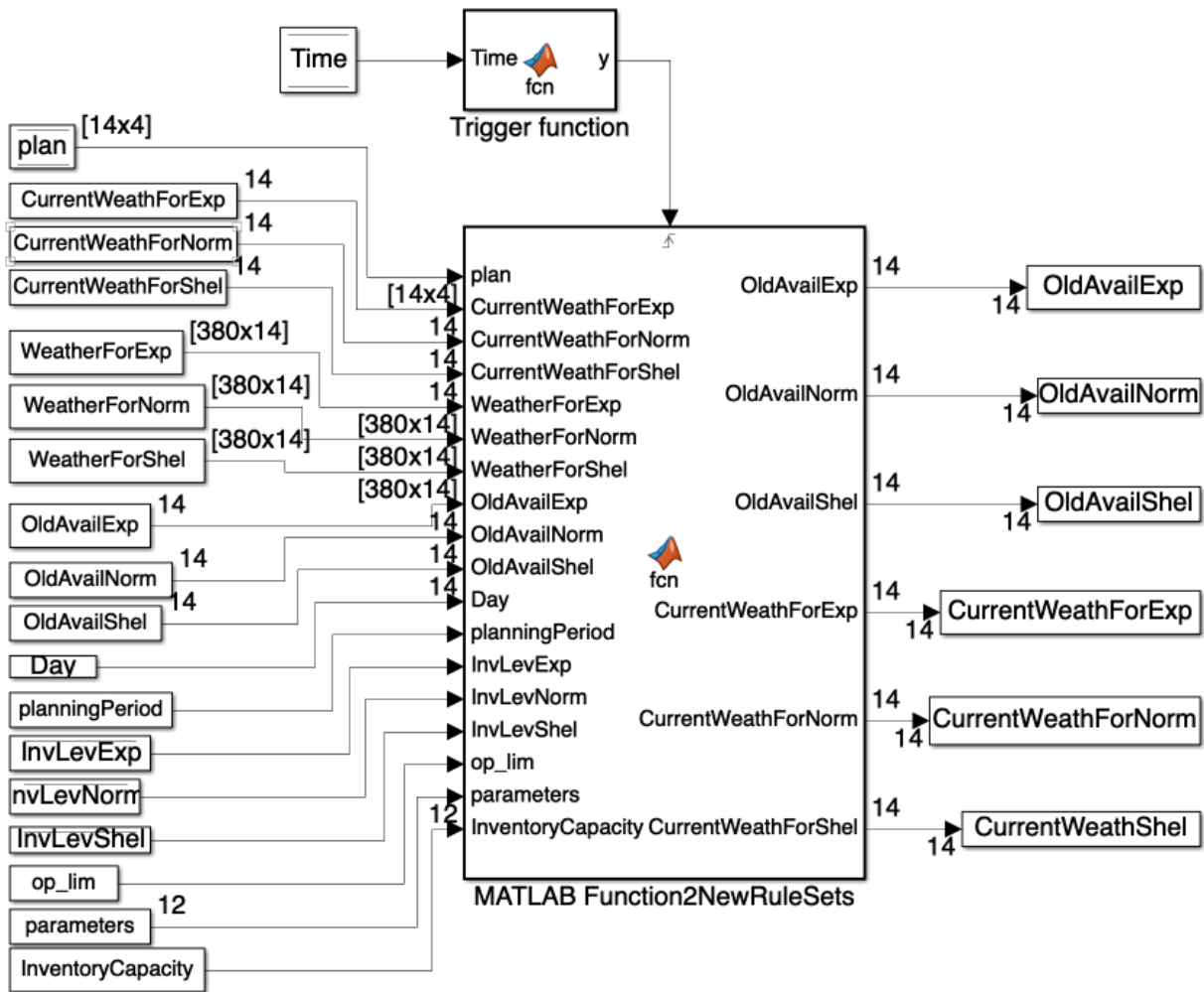


Figure F.3: MATLAB function used for rerouting.

F.3 Global Variables

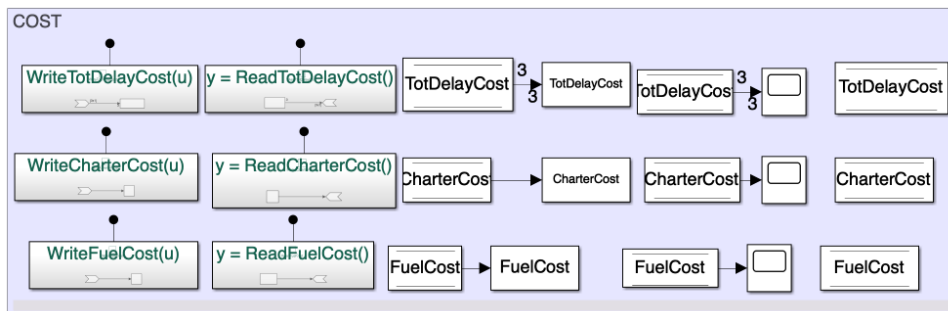


Figure F.4: Global variables for cost components.

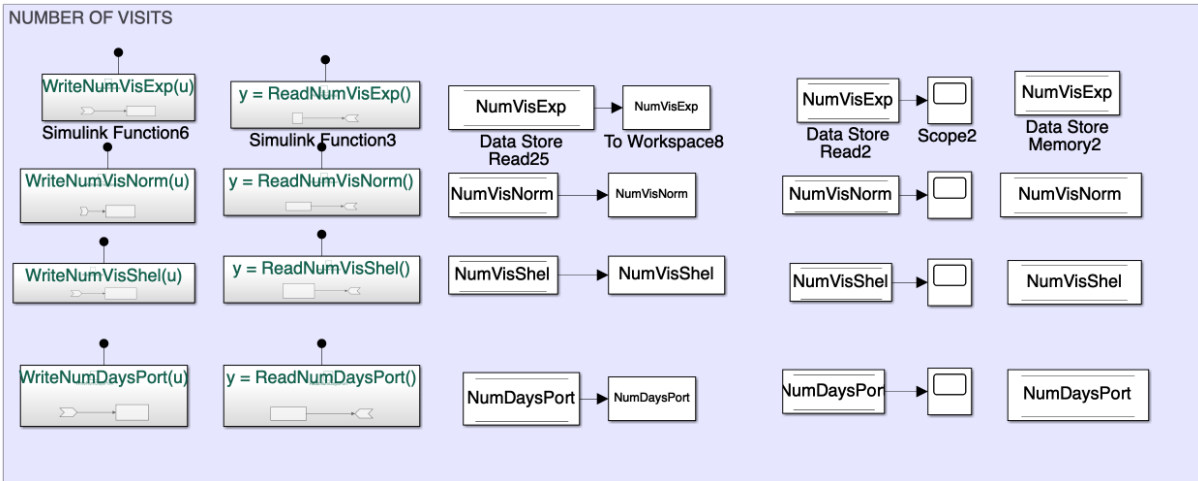


Figure F.5: Global variables for number of visits.

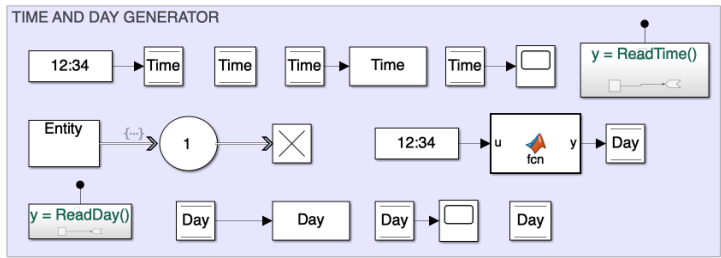


Figure F.6: Time generator with associated global variable.

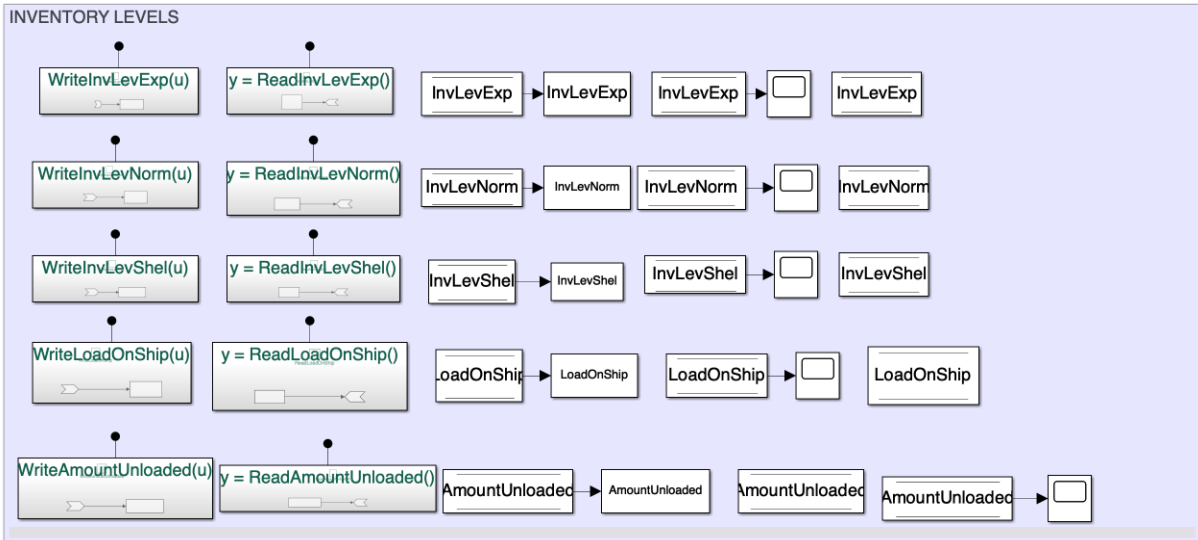


Figure F.7: Global variables for inventory levels.

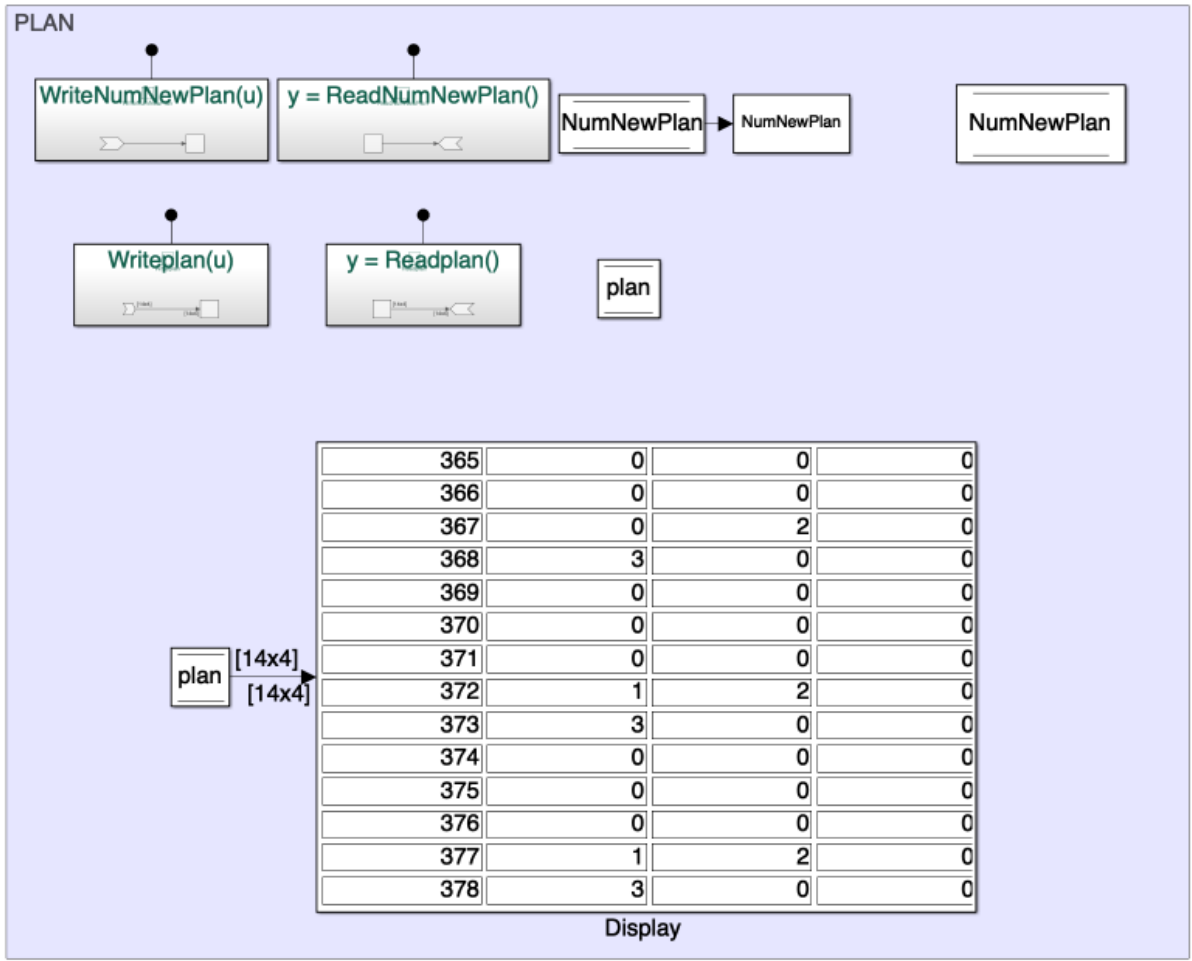


Figure F.8: Plan overview, global variable and display.

G | MATLAB Scripts

G.1 *run_simulation*

```
% This script run the whole DSS model, simulating a year of operation.
% The script imports the data needed from Excel.
% Changes to the data are made in the Excel-document.

clear all
clc
tic

%%% HUSK   OPPDATER TALL I LAGRINGS-FIL!!!! %%%%%%%%%%

% CHOOSE UNCERTAINTY
UncertaintyLevel = input...
('Choose level of uncertainty, 0 = zero, 1 = low, 2 = medium, 3 = high: ')

% SEED. Change between: 12345, 23456, 34567.
Seed = input('Choose seed between 12345, 23456, 34567: ')

% Import all needed parameters/numbers
ship_att = xlsread('ShipAttributes.xlsx');
op_lim = ship_att(3,1);
planningPeriod = 14;
Distances_matrix = xlsread('Distances.xlsx');

% Generate weather forecast. Variables saved in workspace.
run WeatherMatrixOneYear;

% Assign current weather forecast as Day 1 in the first row
% First column is todays weather, the rest is weather forecast next 13 days
CurrentWeathForExp = WeatherForExp(1,:);
CurrentWeathForNorm = WeatherForNorm(1,:);
CurrentWeathForShel = WeatherForShel(1,:);
```

```

% Check availability
run availabilityWeatherMatrix; % Output OldAvailExp etc.

% Import current inventory levels and make a prediction of the inventory
% levels the next two weeks
parameters = xlsread('Parameters.xlsx');
StartInvLevExp = parameters(1);
StartInvLevNorm = parameters(2);
StartInvLevShel = parameters(3);
rate = parameters(4);
LowLimExp = parameters(5);
LowLimNorm = parameters(6);
LowLimShel = parameters(7);
InventoryCapacity = 2000;
DelayCostExpRate = parameters(8);
DelayCostNormRate = parameters(9);
DelayCostShelRate = parameters(10);
FuelCostRate = parameters(11);
CharterCostRate = parameters(12);
count = 0; % Set value so that same script can be used here and Simulink
Day = 1;

% RUN CHOSEN RULE SET

% run Ruleset111;    ruleset = 1;    % No information, just rate
% run Ruleset111;    ruleset = 2;    % Weather next day
% run Ruleset113;    ruleset = 3;    % Weather next weeks
  run Ruleset114;    ruleset = 4;    % Inventory levels
% run Test_plan;    ruleset = 0;    % Used to test model/plan

% Write the plan into an Excel-document.
filename = 'routing_plan.xlsx';
writematrix(plan,filename);
plan

% Run the simulation-model in Simulink to test the rule-set.
%model = 'sim_model1.slx';
model = 'FinalModelNewRuleSets2.slx'; % Old backup before day change.

sim(model)

%% %%%%%%%%%%% OUTPUT %%%%%%%%%%%
% Want to generate graphs for the data logged in the simulation

% Must first save the simulation results as vectors that can be used in the
% plotting.
Time = ans.Time;

```

```

Day = ans.Day;

%InvLevExp2 = zeros(floor(length(Time)/24),1);
InvLevExp = ans.InvLevExp;
InvLevNorm = ans.InvLevNorm;
InvLevShel = ans.InvLevShel;
AmountUnloaded = ans.AmountUnloaded;
LoadOnShip = ans.LoadOnShip;
NumVisExp = ans.NumVisExp;
NumVisNorm = ans.NumVisNorm;
NumVisShel = ans.NumVisShel;
NumDaysPort = ans.NumDaysPort;
NumNewPlan = ans.NumNewPlan;
FinalNumNewPlan = NumNewPlan(length(NumNewPlan));
FuelCost = ans.FuelCost;
FinalFuelCost = FuelCost(length(FuelCost));
TotDelayCost = ans.TotDelayCost;
DelayCostExp = TotDelayCost(length(TotDelayCost),1);
DelayCostNorm = TotDelayCost(length(TotDelayCost),2);
DelayCostShel = TotDelayCost(length(TotDelayCost),3);
CalcTotDelCost = DelayCostExp + DelayCostNorm + DelayCostShel;
CharterCost = ans.CharterCost;
FinalCharterCost = CharterCost(length(CharterCost));
FinalNumbers = [NumVisExp(length(NumVisExp)) ...
    NumVisNorm(length(NumVisNorm)) ...
    NumVisShel(length(NumVisShel))];
TotalNumVis = NumVisExp(length(NumVisExp)) + ...
    NumVisNorm(length(NumVisNorm)) + ...
    NumVisShel(length(NumVisShel));
TotDaysInPort = NumDaysPort(length(NumDaysPort));
FinalDelayCosts = [DelayCostExp DelayCostNorm DelayCostShel];
FinalCosts = [FinalCharterCost FinalFuelCost CalcTotDelCost];
TotCost = FinalCharterCost + FinalFuelCost + CalcTotDelCost;

% Make the variables dependent on the Day instead of hour.
% Only for thos variables being plotted vs time
Day2 = zeros(floor(length(Time)/24),1);
for i = 24:24:(length(Time)-1)
    Day2((i/24)) = Time(i+1)/24; % One hour longer due to zero at first
    InvLevExp2((i/24)) = InvLevExp(i);
    InvLevNorm2((i/24)) = InvLevNorm(i);
    InvLevShel2((i/24)) = InvLevShel(i);
    LoadOnShip2(i/24) = LoadOnShip(i);
    NumVisExp2(i/24) = NumVisExp(i);
    NumVisNorm2(i/24) = NumVisNorm(i);
    NumVisShel2(i/24) = NumVisShel(i);
    NumDaysPort2(i/24) = NumDaysPort(i);
    AmountUnloaded2(i/24) = AmountUnloaded(i);
end

```

```

% Remove values less than zero in inventory levels

for i = 1:length(InvLevExp2)
    if InvLevExp2(i) < 0
        InvLevExp2(i) = 0;
    end

    if InvLevNorm2(i) < 0
        InvLevNorm2(i) = 0;
    end

    if InvLevShel2(i) < 0
        InvLevShel2(i) = 0;
    end

end

% OVERVIEW DELAY COST FOR ALL LOCATIONS
figure(2)
x = categorical({'Exposed', 'Normal', 'Sheltered'});
x = reordercats(x, {'Exposed', 'Normal', 'Sheltered'});
b = bar(x, FinalDelayCosts, 'FaceColor', 'flat');
%legend(b, 'Exposed', 'Normal', 'Sheltered', 'Location', 'northeast')
%title('Delay costs for each location')
ylabel('Cost [NOK]')
set(gca, 'FontSize', 22)

% Change colors for all the three bars.
b.CData(x(1), :) = [0.9290 0.6940 0.1250];
b.CData(x(2), :) = [0.9290 0.6940 0.1250];
b.CData(x(3), :) = [0.9290 0.6940 0.1250];
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'Delaycost'), 'png');

% OVERVIEW COSTS IN GENERAL
figure(3)
c = categorical({'Charter cost', 'Fuel cost', 'Delay cost'});
c = reordercats(c, {'Charter cost', 'Fuel cost', 'Delay cost'});
a = bar(c, FinalCosts, 'FaceColor', 'flat');
title('Total cost components')
ylabel('Cost [NOK]')
set(gca, 'FontSize', 22)

% Change colors for all the three bars.
a.CData(c(1), :) = [0 0.4470 0.7410];
a.CData(c(2), :) = [0.8500 0.3250 0.0980];

```

```

a.CData(c(3), :) = [0.9290 0.6940 0.1250];
%legend(a, 'Charter cost', 'Fuel cost', 'Delay cost', 'Location', 'northeast');
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'Totcost'), 'png');

% OVERVIEW OF NUMBER OF VISITS AT EACH LOCATION
figure(4)
X = categorical({'Exposed', 'Normal', 'Sheltered'});
X = reordercats(X, {'Exposed', 'Normal', 'Sheltered'});
d = bar(X, FinalNumbers);
%legend(d, 'Exposed', 'Normal', 'Sheltered', 'Location', 'northeast')
title('Number of visits to each location')
ylabel('Number of visits')
set(gca, 'FontSize', 22)

% Change colors for all the three bars.
d.CData(X(1), :) = [0 0.4470 0.7410];
d.CData(X(2), :) = [0.8500 0.3250 0.0980];
d.CData(X(3), :) = [0.9290 0.6940 0.1250];
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'Numvis'), 'png');

% DAYS IN PORT VS DAYS IN USE
figure(5)
g = categorical({'Days in operation', 'Days in port'});
g = reordercats(g, {'Days in operation', 'Days in port'});
d = bar(g, [TotalNumVis, TotDaysInPort]);
title('Days vessel is in operation')
ylabel('Number of days')
set(gca, 'FontSize', 22)

% Change colors for all the three bars.
d.CData(X(1), :) = [0 0.4470 0.7410];
d.CData(X(2), :) = [0.8500 0.3250 0.0980];
d.CData(X(3), :) = [0.9290 0.6940 0.1250];
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'Daysop'), 'png');

% CHANGE IN INVENTORY LEVEL FOR EACH LOCATION
figure(6)
plot(Day2, InvLevExp2)
title('Inventory Level Exposed Location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca, 'FontSize', 22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';

```

```

saveas(gca,fullfile(fpath,'Invlevexp'),'png');

figure(7)
plot(Day2,InvLevNorm2)
title('Inventory Level Normal Location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca,'FontSize',22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca,fullfile(fpath,'Invlevnorm'),'png');

figure(8)
plot(Day2,InvLevShel2)
title('Inventory Level Sheltered Location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca,'FontSize',22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca,fullfile(fpath,'Invlevshel'),'png');

% INVENTORY LEVELS PLOTET TOGETHER
figure(9)
y1 = plot(Day2,InvLevExp2);
title('Combined plot with all inventory levels')
xlabel('Time [days]') % Sould change to days
ylabel('Inventory level [ton]')
legend(y1,'Exposed location','Location','southeast')
set(gca,'FontSize',22)

hold on
y2 = plot(Day2, InvLevNorm2);
legend([y1; y2],'Exposed location','Normal location')
y3 = plot(Day2, InvLevShel2);
legend([y1 y2 y3],'Exposed location','Normal location','Sheltered location')

hold off
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca,fullfile(fpath,'combinvlev'),'png');

% AMOUNT UNLOADED/TRANSPORTED BY SHIP
figure(10)
h = histogram(AmountUnloaded2, 'BinWidth', 300);
title('Histogram of amount transported each day')
xlabel('Amount unloaded [ton]') % should change to days

```

```

ylabel('Frequency of this loading amount')
set(gca,'FontSize',22)
xticks([300 600 900 1200 1500 1800 2100 2400 2700 3000]);
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca,fullfile(fpath,'amounttrans'),'png');

% If i want a figure divided into four subplots
figure(11)
subplot(2,2,1)
plot(Day2,InvLevExp2)
title('Inventory level exposed location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca,'FontSize',22)

subplot(2,2,2)
plot(Day2,InvLevNorm2)
title('Inventory level normal location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca,'FontSize',22)

subplot(2,2,3)
plot(Day2,InvLevShel2)
title('Inventory level sheltered location')
xlabel('Time [days]') % should change to days
ylabel('Inventory level [ton]')
set(gca,'FontSize',22)

subplot(2,2,4);
plot(Day2,AmountUnloaded2)
title('Amount unloaded from ship per day')
xlabel('Time [days]') % should change to days
ylabel('Amount [ton]')
set(gca,'FontSize',22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca,fullfile(fpath,'invlevall'),'png');

%%%%%% SEA STATES %%%%%%
% Want to plot sea state vs day. Red dotted line = operational limit
% Exposed
figure(12)
SeaStatesExpWeather = RealWeatherExposed(1:365);

```

```

plot(Day2, SeaStatesExpWeather)
yline(op_lim, '-.r')
xlabel('Time [days]') % should change to days
ylabel('Sea state')
set(gca, 'FontSize', 22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'seastatesexp'), 'png');

% Normal
figure(13)
SeaStatesNormWeather = RealWeatherNormal(1:365);
plot(Day2, SeaStatesNormWeather)
yline(op_lim, '-.r')
xlabel('Time [days]') % should change to days
ylabel('Sea state')
set(gca, 'FontSize', 22)
fpath = '/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/Test';
saveas(gca, fullfile(fpath, 'seastatenorm'), 'png');

%%%%%%%% OUTPUT %%%%%%%%%

fprintf('----- SIMULATION OUTPUT -----\n')
fprintf('
                                     \n')
fprintf('Case number:                %d\n', UncertaintyLevel+1)
fprintf('Rule set nr:                   %d\n', ruleset)
fprintf('Seed:                          %d\n', Seed)
fprintf('Total cost:                    %d NOK\n', TotCost)
fprintf('Charter cost:                  %d NOK\n', FinalCharterCost)
fprintf('Fuel cost:                     %d NOK\n', FinalFuelCost)
fprintf('Delay cost:                   %d NOK\n', CalcTotDelCost)
fprintf('Total number of trips:         %d\n', TotalNumVis)
fprintf('Days in port:                 %d\n', TotDaysInPort)
fprintf('Visits to exposed location:    %d\n', ...
      NumVisExp(length(NumVisExp)))
fprintf('Visits to normal location:     %d\n', ...
      NumVisNorm(length(NumVisNorm)))
fprintf('Visits to sheltered location:  %d\n', ...
      NumVisShel(length(NumVisShel)))
fprintf('Number of days simulated:      %d\n', Day2(length(Day2)))
fprintf('Number of new plans generated: %d\n', FinalNumNewPlan)
fprintf('
                                     \n')

toc

```

G.2 *WeatherMatrixOneYear*

```
% GENERATE WEATHER MATRIX WITH REAL WEATHER AND FORECAST FOR 1 YEAR

% This script import the weather data and generate the transition matrix.
% The real weather for one year is generated using the transition matrix
% and Markov Chains.
% Weather forecast for each day is generated by including uncertainty.

tic % Start a stopwatch timer to measure time
clc

%% Import weather data and generate transition matrix
% Import weather data for one year from an nc file. swh = significant wave
% height.
WeatherData = ncread('WeatherData.nc','swh');
WeatherData = ncread('Frohavet.nc','swh');

% Fetch the wave heights.
Hs = squeeze(WeatherData(1,1,:));

% Choose number of states based on absorbing state and state range
NumStates = 8;

% Find the upper limit of the Hs values and divide the values into even
% groups
uplim = max(Hs);
% Find the state range
stateRange = uplim / NumStates;
% Generate the state values, the values separating the states
stateValues = stateRange:stateRange:uplim;
% Preallocates the matrix showing the state of each data point
HsState = zeros(length(Hs),1);

% Find the state for each data point
for i = 1:length(Hs) % For each data point
    for j = 1:NumStates % For each state
        if Hs(i) <= stateValues(j) % Data point in state j
            HsState(i) = j; % Data point categorized
            break % Exit the state-loop and go to next data point
        end
    end
end

% Find the transition between sea states.
transitions = zeros(NumStates); % Preallocates matrix
for t = 1:length(HsState) - 1
```

```

    % Add transitions at correct place.
    transitions(HsState(t), HsState(t+1)) = transitions(HsState(t),...
        HsState(t+1)) + 1;
end

T = transitions;
% Normalize the transition matrix, row by row
for i = 1:NumStates
    T(i,:) = T(i,:) / sum(T(i,:));
end

% Check to see if there are any absorbing states, T(i,j)=1, where i=j
absorbState = zeros(NumStates);
for i = 1:NumStates
    for j = 1:NumStates
        if T(i,j) == 1
            absorbState(i,j) = absorbState(i,j) + 1;
        end
    end
end

% Want to abort function if absorbing states are present.
if sum(sum(absorbState)) >= 1
    error('Run stopped. Absorbing states. Reduce number of states.');
```

```

%% A matrix for the real weather is generated

numDays = 365+15;
numTrans = numDays * 8;

% Random number seed. Control the random number generator which will be
% used later in the code, such as RAND, RANDI. Use this so that RANDI
% produce a predictable sequence of numbers.
rng(Seed);

% Preallocate a vector for the sea states
SeaStates = zeros(1,numTrans);

% Set random starting sea state
state = randi(NumStates);

% Preallocate a vector to save the state it transition to
SavedStates = zeros(numTrans,1);

% Go through a number of transitions and generate a vector with seastates
% generated with the transition matrix
for i = 1:numTrans
    r = rand(); % Generate value in range 0-1
```

```

for j = 1:NumStates
    prob = 0;
    % Find the current probability of changing to sea state k
    for k = 1:j
        prob = prob + T(state,k);
    end

    if r <= prob % Then the new sea state is j
        state = j; % New state
        SavedStates(i) = j; % Store the state we transition to
        break
    end
    % Move the SeaStates(i) = state; from here
end
SeaStates(i) = state;
end

% Want to order the data in a matrix, rows = days, columns = hours
% 8 data points per day: 00:00, 03:00, 06:00, 09:00, 12:00, 15:00, 18:00,
% 21:00
Matrix = (reshape(SeaStates, [8,numTrans/8]));

% Must transpose the matrix to get correct size and placement.
SeaStateMatrix = transpose(Matrix);

% Scale the weather forecast for the three locations

% EXPOSED
WeatherForExp = SeaStateMatrix;

% NORMAL
SeaStatesNorm = zeros(1,numTrans);
for i = 1:numTrans
    if SeaStates(i) == 1
        SeaStatesNorm(i) = 1;
    elseif SeaStates(i) == 2
        SeaStatesNorm(i) = 2;
    elseif SeaStates(i) > 2 && SeaStates(i) <= NumStates
        SeaStatesNorm(i) = SeaStates(i) - 2;
    end
end
WeatherForNorm = transpose(reshape(SeaStatesNorm, [8,numTrans/8]));

% SHELTERED
SeaStatesShel = zeros(1,numTrans);
for i = 1:numTrans
    SeaStatesShel(i) = 1;
end
RealWeatherMatrixShel = transpose(reshape(SeaStatesShel, [8,numTrans/8]));

```

toc

```
% Fetch the sea state at 12:00 at each location, generate vector.
RealWeatherExposed = WeatherForExp(:,5);
RealWeatherNormal = WeatherForNorm(:,5);
RealWeatherSheltered = RealWeatherMatrixShel(:,5);

%% Generate weather forecast columns based on real weather

% Preallocate matrixes
WeatherForExp = zeros(numDays,14);
WeatherForNorm = zeros(numDays,14);
WeatherForShel = zeros(numDays,14);

% Set column 1 as the real weather
WeatherForExp(:,1) = RealWeatherExposed;
WeatherForNorm(:,1) = RealWeatherNormal;
WeatherForShel(:,1) = RealWeatherSheltered;

% Want to include the possibility of choosing perfect weather or two levels
% of uncertainty. Use an if-sentence to choose.

% Import weather to matrix
for i = 1:365           % Rows
    for j = 1:13       % Columns
        WeatherForExp(i,j+1) = WeatherForExp(i+j,1);
        WeatherForNorm(i,j+1) = WeatherForNorm(i+j,1);
        WeatherForShel(i,j+1) = WeatherForShel(i+j,1);
    end
end

% PERFECT WEATHER
if UncertaintyLevel == 0
    % Sea state for all locations = 1 all the time, always available

    for i = 1:365
        for j = 1:14
            WeatherForExp(i,j) = 1;
            WeatherForNorm(i,j) = 1;
            WeatherForShel(i,j) = 1;
        end
    end

% PERFECT WEATHER FORECAST
elseif UncertaintyLevel == 1
    % Nothing is changed in the matrix.

% WEATHER FORECAST WITH SOME UNCERTAINTY
```

```

elseif UncertaintyLevel == 2

    for i = 1:365
        p = rand(); % Decides if seastate should increase,decrease,stay
        f = rand()*i/10;% Multiplies with "i",increases with distant future
        b = rand(); % Multiplies with "i", increase possib. in future

        for j = 1:13
            if p <= f % Change in sea state

                % Now say it is a 50/50 chance for increase/decrease
                if b >= 0.5 % Increased sea state
                    % Must make sure we are within valied sea states
                    if WeatherForExp(i+j,j) + round(f,0) <= NumStates
                        WeatherForExp(i,j+1) = WeatherForExp(i+j,j) +...
                            round(rand()*j/5,0);
                        WeatherForNorm(i,j+1) = WeatherForNorm(i+j,j)+...
                            round(rand()*j/5,0);
                    end
                else
                    % Make sure within valid sea states
                    if WeatherForExp(i+j,j) + round(f,0) >= 1
                        WeatherForExp(i,j+1) = WeatherForExp(i+j,j) - ...
                            round(rand()*j/5,0);
                        WeatherForNorm(i,j+1) = WeatherForNorm(i+j,j) - ...
                            round(rand()*j/5,0);
                    end
                end
            end
            % Else, stay the same, no change in sea state
        end
    end

% WEATHER FORECAST WITH HIGH UNCERTAINT
elseif UncertaintyLevel == 3

    for i = 1:365
        for j = 1:13
            p = rand(); % Decides if seastate should increase,decrease,stay
            f = rand()*j/5; % Multiplies with "i", increases with future
            b = rand(); % Multiplies with "i", increase possib. in future

            if p <= f % Change in sea state

                % Now say it is a 50/50 chance for increase/decrease
                if b >= 0.5 % Increased sea state
                    % Must make sure we are within valied sea states
                    if WeatherForExp(i+j,j) + round(f,0) <= NumStates

```

```

        WeatherForExp(i, j+1) = WeatherForExp(i+j, j) + ...
            round(rand()*j/5, 0);
        WeatherForNorm(i, j+1) = WeatherForNorm(i+j, j) + ...
            round(rand()*j/5, 0);
    end
else
    % Make sure within valid sea states
    if WeatherForExp(i+j, j) + round(f, 0) >= 1
        WeatherForExp(i, j+1) = WeatherForExp(i+j, j) - ...
            round(rand()*j/5, 0);
        WeatherForNorm(i, j+1) = WeatherForNorm(i+j, j) - ...
            round(rand()*j/5, 0);
    end
end
end
    % Else, stay the same, no change in sea state
end
end

end

% Should make an overview of the Hs distribution for the sea states
binranges = [0 stateValues];
[bincounts] = histc(Hs, binranges);
pdf = bincounts/24480;
figure(1)
g = bar(binranges, bincounts, 'histc');

%title('Histogram of sea states');
xlabel('Sea states') % should change to days
ylabel('Number of data points')
set(gca, 'FontSize', 22)
set(g, 'FaceColor', [0 0.4470 0.7410]);
xticks([1 2 3 4 5 6 7 8]);

fpath='/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/General';
saveas(gca, fullfile(fpath, 'Hsdistribution'), 'png');

figure(22)
pd = bar(binranges, pdf, 'histc');
set(pd, 'FaceColor', [0 0.4470 0.7410]);
set(gca, 'FontSize', 22)
xlabel('Sea states') % should change to days
ylabel('Probability')
xticks([1 2 3 4 5 6 7 8]);

fpath='/Users/Carina/Dropbox/Apps/Overleaf/Master :)/MasterThesis/General';
saveas(gca, fullfile(fpath, 'Hspdf'), 'png');

```

G.3 *availabilityWeatherMatrix*

```
% Check availability

% Compare weather with operational limits - set up plan of availability.
% A vector with 1 if available, 0 if not.
OldAvailExp = zeros(1,planningPeriod);
OldAvailNorm = zeros(1,planningPeriod);
OldAvailShel = zeros(1,planningPeriod);

for i = 1:planningPeriod
    if CurrentWeathForExp(i) < op_lim
        OldAvailExp(i) = 1;
    end

    if CurrentWeathForNorm(i) < op_lim
        OldAvailNorm(i) = 1;
    end

    if CurrentWeathForShel(i) < op_lim
        OldAvailShel(i) = 1;
    end
end
```

G.4 *MATLAB Function2NewRuleSets*

```
function [OldAvailExp,OldAvailNorm, OldAvailShel, CurrentWeathForExp, ...
    CurrentWeathForNorm, CurrentWeathForShel] = fcn( plan, ...
    CurrentWeathForExp, CurrentWeathForNorm, CurrentWeathForShel, ...
    WeatherForExp, WeatherForNorm, WeatherForShel, OldAvailExp, ...
    OldAvailNorm, OldAvailShel, Day, planningPeriod, InvLevExp, ...
    InvLevNorm, InvLevShel, op_lim, parameters,InventoryCapacity)

% Simulation model with new weather matrix!

% 1. Figure out where we are in plan
% 2. Compare current weather forecast with forecast for this day. Get the
% weather forecast from big matrix.
% 3. If predicted worse availability the next week, update current forecast
% and make a new plan based on this.

% Get values from parameters-matrix

rate = parameters(4);
LowLimExp = parameters(5);
```

```

LowLimNorm = parameters(6);
LowLimShel = parameters(7);
CountDay = 1;

Day
% Must check where in the plan we are at this time
if Day == 1
    count = 1;
    CountDay = 1;
else
    % Start over again and find new count each round.
    count = 1;
for i = 1:length(plan)
    if plan(i,1) == Day
        count = i;
        CountDay = i;
        break
    end
end
%end

% If at end of current plan, must update.
if count == 14
    count = 1;
    c = 14;
    % Must get new weather forecast.
    NewForecast = 1;
else
    % Use same weather forecast as before to check with an updated one.
    NewForecast = 0;
    c = 0;
end

%% Fetch a new basis forecast if at end of plan.
if NewForecast == 1
    % Fetch new forecast for given day from weather matrix

    for i = 1:14
        CurrentWeathForExp(i) = WeatherForExp(Day,i);
        CurrentWeathForNorm(i) = WeatherForNorm(Day,i);
        CurrentWeathForShel(i) = WeatherForShel(Day,i);
    end

% Generate availability vector
OldAvailExp = zeros(14,1);
OldAvailNorm = zeros(14,1);
OldAvailShel = zeros(14,1);

for i = count:planningPeriod

```

```

% Must make sure it only checks where there are values, not zeros

if CurrentWeathForExp(i) < op_lim
    OldAvailExp(i) = 1;
end

if CurrentWeathForNorm(i) < op_lim
    OldAvailNorm(i) = 1;
end

if CurrentWeathForShel(i) < op_lim
    OldAvailShel(i) = 1;
end

end
end

%% Get updated forecast for current day from matrix

% Check values
CurrentWeathForExp;
CurrentWeathForNorm;

% Fetch updated forecast from matrix
UpdatedWeathForExp = WeatherForExp(Day,:);
UpdatedWeathForNorm = WeatherForNorm(Day,:);
UpdatedWeathForShel = WeatherForShel(Day,:);

%% Check availability of updated forecast, for this Day

UpdatedAvailExp = zeros(1,planningPeriod);
UpdatedAvailNorm = zeros(1,planningPeriod);
UpdatedAvailShel = zeros(1,planningPeriod);

for i = count:planningPeriod

    if UpdatedWeathForExp(i) < op_lim
        UpdatedAvailExp(i) = 1;
    end

    if UpdatedWeathForNorm(i) < op_lim
        UpdatedAvailNorm(i) = 1;
    end

    if UpdatedWeathForShel(i) < op_lim
        UpdatedAvailShel(i) = 1;
    end

end

end

```

```

%% Should a new plan be made? Compare current and updated forecast
NewPlan = 0;

if count > 7
    avail_check = 14;
else
    avail_check = count + 7;
end

for i = count:avail_check
% Must compare the availability, if worse a new plan must be generated.

    if OldAvailExp(i) == 1
        if UpdatedAvailExp(i) == 0
            NewPlan = 1;
            break % Break the loop once a reason for new plan is found
        else % Updated plan is also available
            NewPlan = 0;
        end
    else % Updated plan is better
        NewPlan = 0;
    end

    % Must make a new plan if we are on the last day
    if c == length(plan)
        NewPlan = 1;
        break
    end
end
NewPlan

%% Generate new plan if NewPlan == 1
% Import current inventory levels and make a prediction of the inventory
% levels the next two weeks

if NewPlan == 1
    % Update current weather
    for i = 1:planningPeriod
        CurrentWeathForExp(i) = UpdatedWeathForExp(i);
        CurrentWeathForNorm(i) = UpdatedWeathForNorm(i);
        CurrentWeathForShel(i) = UpdatedWeathForShel(i);
    end

    % Count number of times the plan is updated
    NumNewPlan = ReadNumNewPlan;
    NumNewPlan = NumNewPlan + 1;
    WriteNumNewPlan(NumNewPlan);

```

```

% Use the current inventory level to check for certain rule sets
StartInvLevExp = InvLevExp;
StartInvLevNorm = InvLevNorm;
StartInvLevShel = InvLevShel;

% Values used to "decode" the model.
actual_values = 1;
count
Day

%% Run chosen ruleset
%% RULE SET 14 SIMULINK - INVENTORY LEVELS

% Preallocates vectors for the prediction of inventory.
InvPredExp = zeros(planningPeriod,1);
InvPredNorm = zeros(planningPeriod,1);
InvPredShel = zeros(planningPeriod,1);

InvPredExp(count) = InvLevExp;
InvPredNorm(count) = InvLevNorm;
InvPredShel(count) = InvLevShel;

% Look at count
if c == 14
    count = 0;
end

% Can only change the future visits, so only go through remaining days of
% the planning period. The count value is found/set earlier.
CurrentWeathForExp;
CurrentWeathForNorm;
UpdatedAvailExp;
UpdatedAvailNorm;

for i = count + 1:planningPeriod
    % Set Day into plan.
    Day;
    plan(i,1) = Day + 1;
    plan(i,2) = 0;
    plan(i,3) = 0;
    plan(i,4) = 0;

    % Generate inventory prediction
    if i == 1 % For the first generation, set start inventory.
        InvPredExp(i) = StartInvLevExp;
        InvPredNorm(i) = StartInvLevNorm;
        InvPredShel(i) = StartInvLevShel;
    else

```

```

    InvPredExp(i) = InvPredExp(i-1) - rate;
    InvPredNorm(i) = InvPredNorm(i-1) - rate;
    InvPredShel(i) = InvPredShel(i-1) - rate;
end

% EXPOSED
% Check if level is beneath lower limit and book visit.
if InvPredExp(i) < LowLimExp
    % Location must be visited during the next days.
    inv_leve_is_lower_than_limit_exp = i;

    % Check availability
    if UpdatedAvailExp(i) == 1
        plan(i,2) = 1;
        InvPredExp(i) = InventoryCapacity;
    end
end

% NORMAL
% Check if level is beneath lower limit and book visit.
if InvPredNorm(i) < LowLimNorm
    % Location must be visited during the next days.
    inv_leve_is_lower_than_limit_norm = i;

    % Check availability
    if UpdatedAvailNorm(i) == 1
        if plan(i,2) == 0
            plan(i,2) = 2;
            InvPredNorm(i) = InventoryCapacity;
        elseif plan(i,3) == 0
            plan(i,3) = 2;
            InvPredNorm(i) = InventoryCapacity;
        end
    end
end

% SHELTERED
% Check if level is beneath lower limit and book visit.
if InvPredShel(i) < LowLimShel
    % Location must be visited during the next days.
    inv_leve_is_lower_than_limit_shel = i;

    % Check availability
    if UpdatedAvailShel(i) == 1
        if plan(i,2) == 0
            plan(i,2) = 3;
            InvPredShel(i) = InventoryCapacity;
        elseif plan(i,3) == 0
            plan(i,3) = 3;

```

```

        InvPredShel(i) = InventoryCapacity;
    end
end
end

Day_value_in_for_loop = 1;
Day = Day + 1;

end % for

% Values used to "check" code
InvPredExp;
%% %%%%%%%%%% END RULE SET %%%%%%%%%%%

% Preallocate vector for availability.
OldAvailExp = zeros(14,1);
OldAvailNorm = zeros(14,1);
OldAvailShel = zeros(14,1);

% Give the values new names to use next round.
for i = 1:planningPeriod
    OldAvailExp(i) = UpdatedAvailExp(i);
    OldAvailNorm(i) = UpdatedAvailNorm(i);
    OldAvailShel(i) = UpdatedAvailShel(i);
end

end % if New plan

Day = ReadDay()
count

%% Make sure operation not completed if bad weather

%% PLACE THIS AFTER NEW PLAN
% Must make sure operation is not completed if location not available
if CountDay < 14
    if plan(CountDay+1,2) ~= 0
        if plan(CountDay+1,2) == 1
            if OldAvailExp(CountDay+1) == 0
                plan(CountDay+1,2) = 0;
            end
        elseif plan(CountDay+1,2) == 2
            if OldAvailNorm(CountDay+1) == 0
                plan(CountDay+1,2) = 0;
            end
        end
    elseif plan(CountDay+1,3) ~= 0

```

```
if plan(CountDay+1,3) == 1
    if OldAvailExp(CountDay+1) == 0
        plan(CountDay+1,3) = 0;
    end

elseif plan(CountDay+1,3) == 2
    if OldAvailNorm(CountDay+1) == 0
        plan(CountDay+1,3) = 0;
    end
end

end
end

%%
% Write the plan into a Data Store, both if new plan is generated and if
% old one is used.
Writeplan(plan);

% Values used to "decode" the model
Day
plan

end
```

H | Simulation Output

Case 1

Rule set 1

```
----- SIMULATION OUTPUT -----
```

Case number:	1	Case number:	1	Case number:	1
Rule set nr:	1	Rule set nr:	1	Rule set nr:	1
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	8096000 NOK	Total cost:	8096000 NOK	Total cost:	8096000 NOK
Charter cost:	1840000 NOK	Charter cost:	1840000 NOK	Charter cost:	1840000 NOK
Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK
Delay cost:	0 NOK	Delay cost:	0 NOK	Delay cost:	0 NOK
Total number of trips:	276	Total number of trips:	276	Total number of trips:	276
Days in port:	182	Days in port:	182	Days in port:	182
Visits to exposed location:	92	Visits to exposed location:	92	Visits to exposed location:	92
Visits to normal location:	92	Visits to normal location:	92	Visits to normal location:	92
Visits to sheltered location:	92	Visits to sheltered location:	92	Visits to sheltered location:	92
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	26	Number of new plans generated:	26	Number of new plans generated:	26

Elapsed time is 42.038231 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.668997 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 33.335260 seconds.

Rule set 2

```
----- SIMULATION OUTPUT -----
```

Case number:	1	Case number:	1	Case number:	1
Rule set nr:	2	Rule set nr:	2	Rule set nr:	2
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	8096000 NOK	Total cost:	8096000 NOK	Total cost:	8096000 NOK
Charter cost:	1840000 NOK	Charter cost:	1840000 NOK	Charter cost:	1840000 NOK
Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK
Delay cost:	0 NOK	Delay cost:	0 NOK	Delay cost:	0 NOK
Total number of trips:	276	Total number of trips:	276	Total number of trips:	276
Days in port:	182	Days in port:	182	Days in port:	182
Visits to exposed location:	92	Visits to exposed location:	92	Visits to exposed location:	92
Visits to normal location:	92	Visits to normal location:	92	Visits to normal location:	92
Visits to sheltered location:	92	Visits to sheltered location:	92	Visits to sheltered location:	92
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	26	Number of new plans generated:	26	Number of new plans generated:	26

Elapsed time is 42.132618 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.884754 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 34.022523 seconds.

Rule set 3

```
----- SIMULATION OUTPUT -----
```

Case number:	1	Case number:	1	Case number:	1
Rule set nr:	3	Rule set nr:	3	Rule set nr:	3
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	8096000 NOK	Total cost:	8096000 NOK	Total cost:	8096000 NOK
Charter cost:	1840000 NOK	Charter cost:	1840000 NOK	Charter cost:	1840000 NOK
Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK	Fuel cost:	6256000 NOK
Delay cost:	0 NOK	Delay cost:	0 NOK	Delay cost:	0 NOK
Total number of trips:	276	Total number of trips:	276	Total number of trips:	276
Days in port:	182	Days in port:	182	Days in port:	182
Visits to exposed location:	92	Visits to exposed location:	92	Visits to exposed location:	92
Visits to normal location:	92	Visits to normal location:	92	Visits to normal location:	92
Visits to sheltered location:	92	Visits to sheltered location:	92	Visits to sheltered location:	92
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	26	Number of new plans generated:	26	Number of new plans generated:	26

Elapsed time is 47.042145 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 42.117311 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.016034 seconds.

Rule set 4

```
----- SIMULATION OUTPUT -----
Case number: 1
Rule set nr: 4
Seed: 12345
Total cost: 8333000 NOK
Charter cost: 2040000 NOK
Fuel cost: 6293000 NOK
Delay cost: 0 NOK
Total number of trips: 205
Days in port: 162
Visits to exposed location: 69
Visits to normal location: 68
Visits to sheltered location: 68
Number of days simulated: 365
Number of new plans generated: 26

Elapsed time is 45.499423 seconds.

----- SIMULATION OUTPUT -----
Case number: 1
Rule set nr: 4
Seed: 23456
Total cost: 8333000 NOK
Charter cost: 2040000 NOK
Fuel cost: 6293000 NOK
Delay cost: 0 NOK
Total number of trips: 205
Days in port: 162
Visits to exposed location: 69
Visits to normal location: 68
Visits to sheltered location: 68
Number of days simulated: 365
Number of new plans generated: 26

Elapsed time is 40.805594 seconds.

----- SIMULATION OUTPUT -----
Case number: 1
Rule set nr: 4
Seed: 34567
Total cost: 8333000 NOK
Charter cost: 2040000 NOK
Fuel cost: 6293000 NOK
Delay cost: 0 NOK
Total number of trips: 205
Days in port: 162
Visits to exposed location: 69
Visits to normal location: 68
Visits to sheltered location: 68
Number of days simulated: 365
Number of new plans generated: 26

Elapsed time is 36.219576 seconds.
```

Case 2

Rule set 1

```
----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 1
Seed: 12345
Total cost: 8615000 NOK
Charter cost: 1990000 NOK
Fuel cost: 6330000 NOK
Delay cost: 295000 NOK
Total number of trips: 248
Days in port: 167
Visits to exposed location: 64
Visits to normal location: 92
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 152

Elapsed time is 31.893352 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 1
Seed: 23456
Total cost: 8572000 NOK
Charter cost: 1940000 NOK
Fuel cost: 6512000 NOK
Delay cost: 120000 NOK
Total number of trips: 263
Days in port: 172
Visits to exposed location: 77
Visits to normal location: 94
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 206

Elapsed time is 34.216352 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 1
Seed: 34567
Total cost: 9588000 NOK
Charter cost: 2190000 NOK
Fuel cost: 7313000 NOK
Delay cost: 85000 NOK
Total number of trips: 272
Days in port: 147
Visits to exposed location: 83
Visits to normal location: 97
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 238

Elapsed time is 35.047626 seconds.
```

Rule set 2

```
----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 2
Seed: 12345
Total cost: 8642000 NOK
Charter cost: 1920000 NOK
Fuel cost: 6497000 NOK
Delay cost: 225000 NOK
Total number of trips: 243
Days in port: 174
Visits to exposed location: 79
Visits to normal location: 82
Visits to sheltered location: 82
Number of days simulated: 365
Number of new plans generated: 152

Elapsed time is 34.701235 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 2
Seed: 23456
Total cost: 9400000 NOK
Charter cost: 2180000 NOK
Fuel cost: 7120000 NOK
Delay cost: 100000 NOK
Total number of trips: 263
Days in port: 148
Visits to exposed location: 85
Visits to normal location: 91
Visits to sheltered location: 87
Number of days simulated: 365
Number of new plans generated: 206

Elapsed time is 36.046327 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 2
Seed: 34567
Total cost: 9262000 NOK
Charter cost: 2070000 NOK
Fuel cost: 7147000 NOK
Delay cost: 45000 NOK
Total number of trips: 275
Days in port: 159
Visits to exposed location: 88
Visits to normal location: 96
Visits to sheltered location: 91
Number of days simulated: 365
Number of new plans generated: 238

Elapsed time is 37.232533 seconds.
```

Rule set 3

```
----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 3
Seed: 12345
Total cost: 7997000 NOK
Charter cost: 1880000 NOK
Fuel cost: 5682000 NOK
Delay cost: 435000 NOK
Total number of trips: 232
Days in port: 178
Visits to exposed location: 50
Visits to normal location: 90
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 152

Elapsed time is 40.271279 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 3
Seed: 23456
Total cost: 8471000 NOK
Charter cost: 1960000 NOK
Fuel cost: 6316000 NOK
Delay cost: 195000 NOK
Total number of trips: 251
Days in port: 170
Visits to exposed location: 65
Visits to normal location: 94
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 206

Elapsed time is 52.511204 seconds.

----- SIMULATION OUTPUT -----
Case number: 2
Rule set nr: 3
Seed: 34567
Total cost: 9594000 NOK
Charter cost: 2190000 NOK
Fuel cost: 7299000 NOK
Delay cost: 105000 NOK
Total number of trips: 270
Days in port: 147
Visits to exposed location: 81
Visits to normal location: 97
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 238

Elapsed time is 43.207176 seconds.
```

Rule set 4

```
----- SIMULATION OUTPUT -----
```

Case number:	2	Case number:	2	Case number:	2
Rule set nr:	4	Rule set nr:	4	Rule set nr:	4
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	7229000 NOK	Total cost:	7042000 NOK	Total cost:	7125000 NOK
Charter cost:	1640000 NOK	Charter cost:	1570000 NOK	Charter cost:	1600000 NOK
Fuel cost:	5129000 NOK	Fuel cost:	5177000 NOK	Fuel cost:	5455000 NOK
Delay cost:	460000 NOK	Delay cost:	295000 NOK	Delay cost:	70000 NOK
Total number of trips:	187	Total number of trips:	196	Total number of trips:	207
Days in port:	202	Days in port:	209	Days in port:	206
Visits to exposed location:	51	Visits to exposed location:	59	Visits to exposed location:	66
Visits to normal location:	67	Visits to normal location:	67	Visits to normal location:	71
Visits to sheltered location:	69	Visits to sheltered location:	70	Visits to sheltered location:	70
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	152	Number of new plans generated:	206	Number of new plans generated:	238

Elapsed time is 39.206954 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 43.415658 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 39.808841 seconds.

Case 3

Rule set 1

```
----- SIMULATION OUTPUT -----
```

Case number:	3	Case number:	3	Case number:	3
Rule set nr:	1	Rule set nr:	1	Rule set nr:	1
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	8686000 NOK	Total cost:	8800000 NOK	Total cost:	9362000 NOK
Charter cost:	1960000 NOK	Charter cost:	2040000 NOK	Charter cost:	2070000 NOK
Fuel cost:	6436000 NOK	Fuel cost:	6675000 NOK	Fuel cost:	7222000 NOK
Delay cost:	290000 NOK	Delay cost:	85000 NOK	Delay cost:	70000 NOK
Total number of trips:	250	Total number of trips:	261	Total number of trips:	274
Days in port:	170	Days in port:	162	Days in port:	159
Visits to exposed location:	65	Visits to exposed location:	75	Visits to exposed location:	84
Visits to normal location:	94	Visits to normal location:	93	Visits to normal location:	97
Visits to sheltered location:	91	Visits to sheltered location:	93	Visits to sheltered location:	93
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	152	Number of new plans generated:	200	Number of new plans generated:	232

Elapsed time is 37.765689 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 42.161009 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.844287 seconds.

Rule set 2

```
----- SIMULATION OUTPUT -----
```

Case number:	3	Case number:	3	Case number:	3
Rule set nr:	2	Rule set nr:	2	Rule set nr:	2
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	8707000 NOK	Total cost:	8987000 NOK	Total cost:	9527000 NOK
Charter cost:	1920000 NOK	Charter cost:	2010000 NOK	Charter cost:	2140000 NOK
Fuel cost:	6527000 NOK	Fuel cost:	6912000 NOK	Fuel cost:	7377000 NOK
Delay cost:	260000 NOK	Delay cost:	65000 NOK	Delay cost:	10000 NOK
Total number of trips:	243	Total number of trips:	266	Total number of trips:	281
Days in port:	174	Days in port:	165	Days in port:	152
Visits to exposed location:	74	Visits to exposed location:	86	Visits to exposed location:	91
Visits to normal location:	82	Visits to normal location:	92	Visits to normal location:	98
Visits to sheltered location:	87	Visits to sheltered location:	88	Visits to sheltered location:	92
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	152	Number of new plans generated:	200	Number of new plans generated:	232

Elapsed time is 35.476246 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 33.263714 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.026666 seconds.

Rule set 3

```
----- SIMULATION OUTPUT -----
```

Case number:	3	Case number:	3	Case number:	3
Rule set nr:	3	Rule set nr:	3	Rule set nr:	3
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	7886000 NOK	Total cost:	8364000 NOK	Total cost:	9392000 NOK
Charter cost:	1820000 NOK	Charter cost:	1890000 NOK	Charter cost:	2070000 NOK
Fuel cost:	5531000 NOK	Fuel cost:	6209000 NOK	Fuel cost:	7252000 NOK
Delay cost:	535000 NOK	Delay cost:	185000 NOK	Delay cost:	70000 NOK
Total number of trips:	230	Total number of trips:	251	Total number of trips:	274
Days in port:	184	Days in port:	177	Days in port:	159
Visits to exposed location:	48	Visits to exposed location:	67	Visits to exposed location:	84
Visits to normal location:	89	Visits to normal location:	93	Visits to normal location:	97
Visits to sheltered location:	93	Visits to sheltered location:	91	Visits to sheltered location:	93
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	152	Number of new plans generated:	200	Number of new plans generated:	232

Elapsed time is 45.746863 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.602982 seconds.

```
----- SIMULATION OUTPUT -----
```

Elapsed time is 37.289587 seconds.

Rule set 4

```
----- SIMULATION OUTPUT -----
Case number: 3
Rule set nr: 4
Seed: 12345
Total cost: 6583000 NOK
Charter cost: 1470000 NOK
Fuel cost: 4700000 NOK
Delay cost: 405000 NOK
Total number of trips: 188
Days in port: 219
Visits to exposed location: 54
Visits to normal location: 64
Visits to sheltered location: 70
Number of days simulated: 365
Number of new plans generated: 152

Elapsed time is 37.476762 seconds.

----- SIMULATION OUTPUT -----
Case number: 3
Rule set nr: 4
Seed: 23456
Total cost: 7235000 NOK
Charter cost: 1700000 NOK
Fuel cost: 5245000 NOK
Delay cost: 290000 NOK
Total number of trips: 195
Days in port: 196
Visits to exposed location: 57
Visits to normal location: 67
Visits to sheltered location: 71
Number of days simulated: 365
Number of new plans generated: 208

Elapsed time is 38.073471 seconds.

----- SIMULATION OUTPUT -----
Case number: 3
Rule set nr: 4
Seed: 34567
Total cost: 7526000 NOK
Charter cost: 1740000 NOK
Fuel cost: 5686000 NOK
Delay cost: 100000 NOK
Total number of trips: 205
Days in port: 192
Visits to exposed location: 64
Visits to normal location: 69
Visits to sheltered location: 72
Number of days simulated: 365
Number of new plans generated: 232

Elapsed time is 38.234782 seconds.
```

Case 4

Rule set 1

```
----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 1
Seed: 12345
Total cost: 8720000 NOK
Charter cost: 1950000 NOK
Fuel cost: 6683000 NOK
Delay cost: 95000 NOK
Total number of trips: 266
Days in port: 171
Visits to exposed location: 78
Visits to normal location: 94
Visits to sheltered location: 94
Number of days simulated: 365
Number of new plans generated: 228

Elapsed time is 34.146724 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 1
Seed: 23456
Total cost: 9113000 NOK
Charter cost: 2000000 NOK
Fuel cost: 6930000 NOK
Delay cost: 95000 NOK
Total number of trips: 271
Days in port: 158
Visits to exposed location: 81
Visits to normal location: 98
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 266

Elapsed time is 35.501757 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 1
Seed: 34567
Total cost: 8722000 NOK
Charter cost: 1980000 NOK
Fuel cost: 6702000 NOK
Delay cost: 40000 NOK
Total number of trips: 266
Days in port: 168
Visits to exposed location: 80
Visits to normal location: 90
Visits to sheltered location: 96
Number of days simulated: 365
Number of new plans generated: 253

Elapsed time is 36.287881 seconds.
```

Rule set 2

```
----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 2
Seed: 12345
Total cost: 8863000 NOK
Charter cost: 1970000 NOK
Fuel cost: 6848000 NOK
Delay cost: 45000 NOK
Total number of trips: 267
Days in port: 169
Visits to exposed location: 85
Visits to normal location: 95
Visits to sheltered location: 87
Number of days simulated: 365
Number of new plans generated: 228

Elapsed time is 45.895947 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 2
Seed: 23456
Total cost: 9055000 NOK
Charter cost: 2000000 NOK
Fuel cost: 6855000 NOK
Delay cost: 90000 NOK
Total number of trips: 270
Days in port: 158
Visits to exposed location: 84
Visits to normal location: 95
Visits to sheltered location: 91
Number of days simulated: 365
Number of new plans generated: 266

Elapsed time is 42.513604 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 2
Seed: 34567
Total cost: 9368000 NOK
Charter cost: 2000000 NOK
Fuel cost: 7253000 NOK
Delay cost: 35000 NOK
Total number of trips: 276
Days in port: 158
Visits to exposed location: 80
Visits to normal location: 95
Visits to sheltered location: 93
Number of days simulated: 365
Number of new plans generated: 253

Elapsed time is 37.341138 seconds.
```

Rule set 3

```
----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 3
Seed: 12345
Total cost: 8406000 NOK
Charter cost: 1900000 NOK
Fuel cost: 6276000 NOK
Delay cost: 230000 NOK
Total number of trips: 248
Days in port: 176
Visits to exposed location: 66
Visits to normal location: 88
Visits to sheltered location: 94
Number of days simulated: 365
Number of new plans generated: 228

Elapsed time is 37.602521 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 3
Seed: 23456
Total cost: 8770000 NOK
Charter cost: 2030000 NOK
Fuel cost: 6530000 NOK
Delay cost: 210000 NOK
Total number of trips: 254
Days in port: 163
Visits to exposed location: 65
Visits to normal location: 97
Visits to sheltered location: 92
Number of days simulated: 365
Number of new plans generated: 266

Elapsed time is 40.840589 seconds.

----- SIMULATION OUTPUT -----
Case number: 4
Rule set nr: 3
Seed: 34567
Total cost: 7690000 NOK
Charter cost: 1820000 NOK
Fuel cost: 5780000 NOK
Delay cost: 90000 NOK
Total number of trips: 252
Days in port: 184
Visits to exposed location: 74
Visits to normal location: 79
Visits to sheltered location: 99
Number of days simulated: 365
Number of new plans generated: 253

Elapsed time is 43.642411 seconds.
```

Rule set 4

----- SIMULATION OUTPUT -----		----- SIMULATION OUTPUT -----		----- SIMULATION OUTPUT -----	
Case number:	4	Case number:	4	Case number:	4
Rule set nr:	4	Rule set nr:	4	Rule set nr:	4
Seed:	12345	Seed:	23456	Seed:	34567
Total cost:	6979000 NOK	Total cost:	7347000 NOK	Total cost:	7212000 NOK
Charter cost:	1590000 NOK	Charter cost:	1690000 NOK	Charter cost:	1670000 NOK
Fuel cost:	5204000 NOK	Fuel cost:	5407000 NOK	Fuel cost:	5427000 NOK
Delay cost:	185000 NOK	Delay cost:	250000 NOK	Delay cost:	115000 NOK
Total number of trips:	196	Total number of trips:	198	Total number of trips:	204
Days in port:	207	Days in port:	197	Days in port:	199
Visits to exposed location:	60	Visits to exposed location:	57	Visits to exposed location:	65
Visits to normal location:	66	Visits to normal location:	69	Visits to normal location:	68
Visits to sheltered location:	70	Visits to sheltered location:	72	Visits to sheltered location:	71
Number of days simulated:	365	Number of days simulated:	365	Number of days simulated:	365
Number of new plans generated:	228	Number of new plans generated:	266	Number of new plans generated:	253
Elapsed time is 37.914606 seconds.		Elapsed time is 47.474299 seconds.		Elapsed time is 39.712286 seconds.	

