Battery Containers and Charging Stations Optimization and Scheduling for a Single-Port Electrified Inland Shipping System

J.C. van Eig
4643062

August 17, 2023

Supervisor TU Delft: dr. ir. T.M.L. Janssen
Supervisors Quo Mare: msc. W. Kuijsten
msc. T. van den Hoogenband

Acknowledgements

Mathematics for me has always been about solving mightily interesting puzzles. The reward for me was a result, a satisfactory outcome, like winning a game of chess. Annotating the way how I reached this outcome, showing the steps, was always slightly annoying, especially the reporting part. Those steps happened in my head and I was perfectly happy to clarify them orally. But writing it down in my rather illegible handwriting, was another beast. I always blamed my dyslexia for this lack of interest and capability. Therefore, I was never looking forward to writing a thesis, because everyone knows this would involve a lot of writing.

Luckily I learned during my years in Leiden and Delft two important things about this aspect of studying. Firstly, there is really added value in writing down your findings. Science is about standing on the shoulders of people who did earlier research. They showed us in detail how they come up with their finding. This helps enormously in finding your own path. Secondly, and equally important, it is really worthwhile and rewarding to ask for help instead of trying to find everything out yourself.

First of all, I would like to thank my daily supervisor at the company Quo Mare in Gouda, Tijn van den Hoogenband. Without him and without his senior colleagues Wim Kuijsten and Gregor Brandt, I would have never managed to dig so deep into this challenging puzzle. Also, a special thanks to Rutger de Mare, the owner of this outstanding company in the field of operational research. He gave me not only the opportunity to do this research but also to gain valuable real work experience for more than a year.

I would also like to thank my supervisor at the TU Delft, Teun Janssen for his guidance and patience. His practical and quick suggestions kept me going and have ensured the delivery of this piece today. Also, I would like to thank my former supervisor, Krzysztof Postek, for helping me set up the project and making sure there was a smooth transition in his absence.

My parents Koen van Eig and Hetty Romp and my brother Pieter have greatly supported me by giving their unconditional trust that I would finish the job in time. They patiently listened when I was frantically ranting about the difficulties of programming or the wide range of obstacles within this assignment, while tirelessly trying to provide me with practical tips and, at least so important, good and healthy food.

Last and foremost, the biggest praise goes to my girlfriend Lauren. After many years, she knows perfectly well how to deal with a mathematics student. She has endured countless hours of me staring at my screen and answering all her questions with a dismissive yes. She pulled me through the dire times when nothing seemed to work and convinced me that I would find a way. She carefully dissected some of my broken sentences and has always been eager to help. Having taken so much of her time, it is only fitting to dedicate this piece of research to her.

Delft, 2023

Abstract

This thesis introduces a model designed to provide port operators with insights into necessary infrastructure and adequate scheduling approaches when ships use electricity as their main power source. The operator manages a shipping port equipped with charging stations and provides a selection of electricity-powered freight cargo ships with container batteries. The model is designed for an arbitrary single-port waterway system. The focus is directed toward inland waterway systems, a choice influenced by the limited capacity of container batteries. Optional external revenue streams in the form of grid balancing and two core uncertainties of the maritime sector are incorporated into the model, namely energy consumption and arrival time uncertainty. Optimization approaches are formulated on the model to find the optimal number of batteries, charging stations, and grid balancing stints. The following six approaches are employed: approximation algorithm, MIP formulation, rolling horizon, probabilistic constraint, extreme value analysis, and value at risk. The strategies all produce a schedule that guides the operator in managing the infrastructure optimally in a different context. The approaches are tested on a simulated waterway system. The approximation algorithm is a great first step. The MIP formulation provides a valuable next step in insight into the complexities of the system. However, it scales too poorly to extend to bigger data cases or to integrate uncertainties. In scenarios involving uncertainty, the rolling horizon approach is recommended due to its adaptability and realistic modeling. Valuable insights about the limits of the system can be obtained by implementing values at risk and extreme value analyses. The probabilistic constraint approach is a suitable alternative if normally distributed uncertainty is inherent to the uncertainty data. The model with these optimization methods provides port operators with essential insights into the system they are managing.

Notation

Name	Indices	Descriptions
$\mathcal{S}$	s	Set of ships
$\mathcal{G}$	g	Set of grid balancing spots
$\mathcal{B}$	b	Set of batteries
$\mathcal{T}$	t, t_1, t_2	Set of event time-space
$\mathcal{T}_a$	t_a	Set of time events where a ship arrives
$\mathcal{T}_g$	t_g	Set of the time events where a grid balancing spot starts
$t_{\min}$		A virtual starting point at the very start of the timeline
$t_{\max}$		A virtual ending point at the very end of the timeline
C_t^-		The minimum charge for a battery allocated at time t
C_t^+		The maximum charge for a battery allocated at time t
$[C_t^-, C_t^+]$		The interval of charge level that a battery must contain to be allocated at time event t
T_g		The timing of time-event t_g
T_a		The ETA prediction with uncertainty in the arrival a
d		The charging speed if a battery is connected to the grid for charging
e		The self-discharge rate
S_b		The starting position of battery b
c_b^0		The starting charge level of battery b
R_{t_g}		The revenue of grid balancing spot t_g
E_t		The energy level difference generated by the event starting on time t
Δ^t		The time difference between event t and event $t-1$
A_{t_1,t_2}		The matrix that shows the relation of battery availability and consecutive time events (See Appendix B.1)
x_b^t		The decision variable that assigns a battery b at time t
c_b^t		The charge level of the battery b at time t
w_b^t		The decision variable that shows the percentage of time battery b is charging within time frame $[t-1, t]$
y_b		The variable that indicates whether or not battery b is used somewhere in the schedule
B		The number of battery containers used.
M		The number of charging stations used.
$(\mathbf{x},\mathbf{w},\mathbf{c})$		The schedule of the port operator, relays the decisions for allocation and charging
(B, M)		The infrastructure, consists of the batteries in use and the number of charging stations at the charging harbor
U_b^t		A status variable that denotes if battery b is currently utilized by a vessel
I_b^t		A status variable that denotes if battery b is currently idle
β_b^t		The critical value for the probability that the schedule fails using battery b at time event t
β		The lower bound for the critical values such that the schedule fails at any time using any battery
$\hat{A}_{t_1^-,t_2^+}$		The matrix that shows the relation of battery availability and consecutive time events in the value at risk of arrival time uncertainty. (See Appendix B.2)
V_b^t		Violation capacity of a battery
N		Arbitrary large number

Abbreviations

Abbreviation	**Name**	Section
AIS	Automatic identification system	1
BC	Base Case	6.1.1
BC2	Base Case 2	6.2.1
BO	Battery optimization	6.1.2
BPO	Battery priority optimization	6.1.2
BPP	Bin-packing problem	A
BPP-Dec	Decision version of BBP	3.1.1
CFAS	Constant-factor Approximation Scheme	3.2
CSO	Charging station optimization	6.1.2
CSPO	Charging station priority optimization	6.1.2
FP	Feasibility Problem	2.5
FP-Dec	Decision version of FP	3.1.1
DMCS	Direct Monte Carlo simulation	5.2.1
ETA	Estimated time of arrival	1
EVs	Electric vehicles	1
MIP	Mixed linear integer problem	5.1
MOP	Multi-objective optimization problem	4.1
RH	Rolling horizon	5.2.2
RP	Revenue Problem	2.5
RP-Dec	Decision version of RP	3.1.2
RPO	Revenue problem optimization	6.1.2
PC	Probabilistic Constraint	5.2.3
$\mathcal{PF}^*$	Pareto Front	4.1
VAR	Value at Risk	5.3.2
3PP	3-Partition Problem	A

Contents

1 Introduction

The transportation industry heavily relies on combustion engines that consume non-renewable fuels. These engines are one of the main carbon emitters in our ecosystem. CO2 emissions are the most significant single cause of climate change. In every aspect of transportation, effort should be made to replace fossil fuels with alternative fuels that are renewable and sustainable. The maritime industry is one of the largest polluters of all transportation options (IEA [2020]). Typically, marine vessels run on heavy fuel oil (marine gasoil, MGO), which is the cheapest variety of fossil fuels. MGO emits greenhouse gases that participate in the increasing severity of climate change, according to Bengtsson et al. [2011], but also nitrogen oxides (NOx) and fine dust (particulate matter, PM). Inland shipping has the same issues (Yan et al. [2018]), where these emissions directly affect nature and cities alongside the inland waterways. Applying alternative fuels for the marine industry proves to be an even bigger challenge than for the electric automobiles industry (EV-industry), because of the superior energy density of fossil fuels, the well-established, incumbent engine industry, and the highly efficient bunkering/loading infrastructure. A system change is required for this industry to facilitate the transition away from fossil fuels.

A viable alternative energy source for vessels is green electricity. The innovations in battery capacity make it possible to create container-sized batteries that can facilitate the energy needed for marine voyages. These container batteries powering the electric motors of cargo vessels already exist and are used in multiple areas, such as maritime shipping. Worldwide companies[1] are established that provide container batteries specifically for marine use. The market penetration is still extremely limited, in comparison with the EV industry. The maritime sector, like the aviation sector, was exempted from the 2015 Paris Treaty. Only recently the European Union has announced measures to contain the emission of the maritime sector ECS. Electrification can be an important step in making sure the goals of the European Union Commission are met.

There are two fundamentally different ways to operate within an electrified maritime system. The vessels could have their own batteries and wait on these to be recharged as they are harbored. On the other hand, the harbor could facilitate a switch in batteries to reduce the stationary time and anchor spots occupied by the ships. For the latter, harbors should house a certain amount of extra batteries. The charging of the batteries while staying harbored increases the waiting time of the electrified vessels drastically compared to the instant swap of an empty to a fresh battery. This extra waiting time might result in delays for the vessel operators and eventually for their clients. The swapping of batteries relieves them of waiting because the swap can be combined with the usual loading and reloading of the cargo. This operation method creates a lower threshold for vessel owners to be convinced to transition. The battery container provider companies[2] also implement swapping services. Therefore, the swapping of batteries at the charging harbors is considered to be the method of choice. This method is also commercially used as can be seen in Figure 41 in Appendix C.

To charge a battery container, it must be attached to the power grid. Charging container batteries is done by attaching them to a specifically designed charging station. The charging station can be used in two distinct ways. Firstly, as a method of charging a battery that is attached. Secondly, the charging station can use the attached battery to strategically trade electricity with the grid. The second option is called electricity trading and requires a spare battery that can store energy. Electricity trading can generate an additional revenue stream for the operators of the docking

[1] Corvus, Fleetzero, Skoon, Zero Emission Maritime, and Zero Emission Services.

[2] See footnote 1.

stations and/or batteries. This value stacking is important to improve the business case and to reduce the cost gap with traditional fuels. Electricity fluctuates in price throughout the day. When the sun shines, the wind blows, and everybody is at work, the energy will be relatively cheap. However, when it gets dark and everybody is home, the price increases. Energy trading is taking advantage of this price gap, buying in cheap and then selling high (Aaltonen et al. [2022]). Ports do not participate in energy trading as part of their core business. Therefore, they are likely to lease the batteries to energy companies for them to use for grid balancing (net stabilization). Grid balancing is a process where energy providers use electricity storage, such as these battery containers, to either store or release energy to regulate the supply and demand of energy in the whole system. Both these secondary occupations of the battery, in the maritime sense, can be considered as revenue options where a battery and station are occupied for a certain period for a certain price.

This thesis aims to develop a strategy that creates a benchmark of the necessary battery containers and charging stations, enabling the transition towards a new, fully electrified, and therefore zero-emission system. Furthermore, it aims for determining the optimal strategy to schedule the infrastructure taking into account the core uncertainties of the maritime industry.

For a singular vessel, it is evident to keep track of the battery containers and schedule them accordingly to be charged before the next arrival. However, if the operations are scaled up to an entire system of electrified system of cargo vessels, the battery containers may be shared among the vessels. The charging stations could also be shared among the on-site batteries. As these charging stations and batteries are costly, the investments and the usage of this infrastructure should be optimized in the sense that the least number of batteries and charging stations are necessary to support the maritime system. Also, time gaps that the batteries lay idle, should preferably be filled by lending the batteries to a revenue option to maximize the profitability of the infrastructure as described above. This research is limited to a singular docking port that is equipped with charging stations. The system with multiple docking ports includes an imbalance in the locations of the batteries. This is outside of the scope of research.

There is a lot of uncertainty related to the maritime industry. Ksciuk et al. [2022] identify six types of uncertainty that are covered by relevant literature on vessel scheduling problems: sailing time, port duration, spot rate revenue, supply, demand, and weather. These factors have a major impact on decisions regarding cargo flows, vessel capacity, sailing speed, and fuel consumption. Three of these uncertainties will be considered for this research: fuel consumption, sailing time, and port duration. The fuel consumption uncertainty is greatly influenced by the sailing speed, weather, and the mass transported. Due to the electrification of the system, the fuel consumption uncertainty is called energy consumption uncertainty. For simplicity, we combine sailing time and port duration in a singular arrival time uncertainty that is given by an estimated time of arrival (ETA) accompanied by an uncertainty distribution. The weather is not directly included in the research, however, it is implicitly included in the ETA uncertainty and the energy consumption uncertainty. Spot rate revenue, supply, and demand lay outside of the scope of this research because there is only a single harbor that is considered.

In this research, we consider a system of maritime vessels, that all connect to a common docking port, making a transition to fully electric-powered vessels with a list of revenue options (energy trading). All major maritime ships are obliged to carry an automatic identification system (AIS) transponder that can be received by radio towers in the neighborhood. The locations of the ports can be assumed to be fixed since it is very rare that industries make a completely new network to facilitate a transition. There are corporations[3] that collect historical AIS-data. This data can be used to extract a system with the arrival times characteristics of the included vessels. These characteristics can be used to estimate the time uncertainties used in this research.

Battery containers are known to be very heavy. Heavy lifting cranes that satisfy safety protocols and permits are needed on a site to swap these containers. These types of cranes can usually already be found at the ports where cargo is transferred. Therefore, existing docking ports should be viewed as possible candidates to fit with charging stations.

Combining the maritime shipping business with grid trading requires tight planning, due to both businesses relying heavily on their energy supply. Any delay or cancellation can have great consequences. Both businesses get by on tiny margins of profit. This will result in high penalties enforced if agreements are missed. To minimize the risk of missing agreed times, an intricate scheduling problem must be solved and optimized. The schedule has to monitor the batteries very precisely and allocate them accordingly. Then, as much revenue as possible can be earned, while the risk of missing agreements is minimized.

The thesis is organized as follows: In chapter 2, the full problem with scope assumptions is described with a model of the simplified electrified maritime system. In chapter 3, the complexity of the model is discussed, where the mathematical problem and its features are compared to known problems. In chapter 4, Pareto optimality is discussed, which is the concept used to describe conflicting objectives in an optimization problem. In chapter 5, the optimization methods that are used to solve the problem are discussed. In chapter 6, the experimental results of the different case studies are shown. The following three case studies are evaluated:

1. A single port deterministic system with simulated arrival times and fixed energy consumption.
2. A single port system with uncertainty in energy consumption by both vessels and grid balancing stints.
3. A single port system with uncertainty in arrival times of the cargo vessels.

In chapter 7, some inference is made about the case studies, and the benefits and shortcomings of the model representation are discussed with some recommendations for further research.

[3] Cofano, Marine Traffic

2 The Model

An inland waterway system is a network of rivers, lakes, and canals that are used by freight vessels. These vessels are not only connecting the supply and demand via the inland ports and terminals within the system but are also transporting cargo from and to the system via the larger deep-sea ports. In general, vessels will carry cargo more economically than can be done by road. Figure 1 shows an example of a system specifically the system within the Netherlands, possibly the most advanced and intricate inland waterway system in the world, closely connected with neighboring countries Belgium and Germany. Rotterdam (NL), Amsterdam (NL), Moerdijk (NL), Gent (B), and Antwerpen (B) are the main deep and coastal sea ports whereto and from a large part of the cargo will be shipped for international trade. To grasp the core characteristics and parameters of such a system, the system will be simplified and a model for the system will be created.

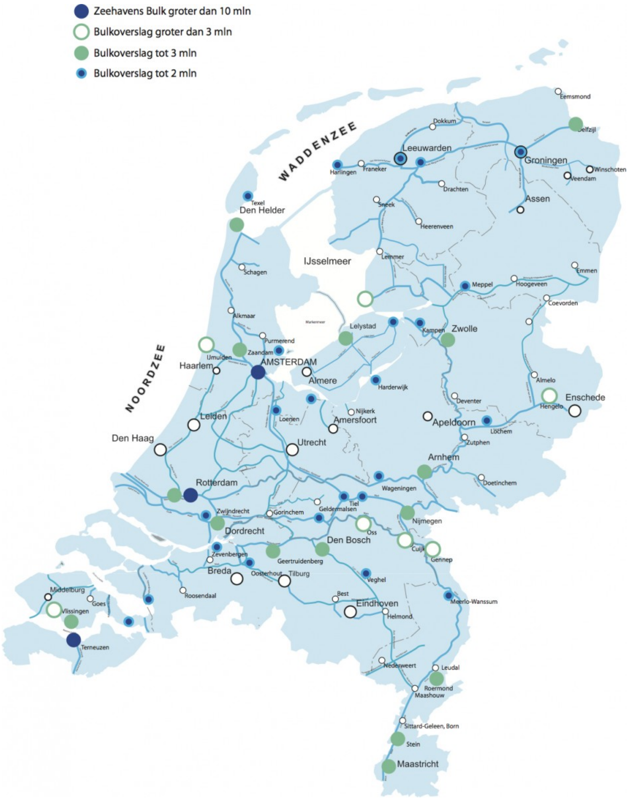


Figure 1: The inland waterway system of the Netherlands with all major harbors and waterways that connect them. Retrieved from BVB website, accessed on 13-08-2023

This chapter describes the problem, the scope, and the mathematical model built to address the problem. First, the full problem of electrifying inland waterways is described. Then, the scope of this research is established. Subsequently, the model is devised to simplify the real-world complexities, with its underlying assumptions are elaborated upon. Then, the model sets, parameters and decision variables utilized in this research is detailed. Then, the constraints on these characteristics are derived. Finally, the optimization goals are described as objective functions in terms of the decision variables of the model.

2.1 Problem Description

We consider the problem of electrifying an inland waterway system from the perspective of harbor operators. A waterway system, as illustrated in Figure 1, consists of a number of ports and cargo vessels. The goal is to achieve total electrification of this system. While growing literature describes the problem of electrifying vehicles, a substantial amount of this literature focuses on regular modes of transportation such as cars, buses, and bikes. Limited literature is dedicated to the electrification of ships, especially within the context of an inland waterway system.

While the goal of electrifying the waterway system is clear, some factors and challenges need to be examined and addressed. According to Muzir et al. [2022], transitioning to electricity in the context of electric vehicles (EVs) faces five main challenges. Perčić et al. [2021] state the same main challenges in the specific context of electric ships. The following challenges are mentioned.

Firstly, the issue of the current high market price for both EVs and batteries. The cost of these components highly influences the adaptation of EVs. Secondly, the limitation of battery capacity, leading to restricted travel distances for EVs. Thirdly, the inadequate charging infrastructure, resulting in the lack of widely available charging stations. Fourthly, the extended charging time, leading to prolonged waiting times. Lastly, the concerns regarding safety and risk, as the technology is relatively new, there exists a raised concern of EVs not being maintained properly. The battery and charging components pose potential electrical, mechanical, and chemical dangers.

Due to the large batteries necessary to propel cargo vessels, some additional challenges emerge such as electricity network congestion for charging stations, reliance on heavy-lifting cranes, and the willingness of the industry. These challenges imply that there may be limited locations and vessels that can be transitioned to electricity.

However, these large batteries also create some opportunities to engage in energy trading. Idle batteries can be used to temporarily store electricity when the prices are low and sell the electricity when the prices are high. The utilization of these trading options may help to offset the high transition costs.

2.1.1 Scope

This research aims to tackle the question of inadequate charging infrastructure for electric freight vessels. This research also only considers a single port in the waterway system that is equipped with charging stations. Furthermore, the research focuses on a system with only freight cargo vessels with predetermined routes that are to be transitioned to using battery containers. Figure 2 shows an example of system scope where five freight vessels sail the system. The battery containers can be considered distinct entities, separate from the freight vessels, and they can serve purposes beyond supplying electricity to freight cargo ships.

Moreover, the existence of predetermined grid balancing stints is assumed, which have an assigned length and revenue. The battery containers can be used on-site to accommodate the grid balancing tasks and add a revenue stream to the charging port at hand.

2.2 Model Description

The problem described in the previous section is modeled to simplify the problem and be able the effectively answer the thesis question. This section is divided into two pieces. First, the assumptions that are made upon the system are listed and explained. Second, the mathematical formulation is derived.

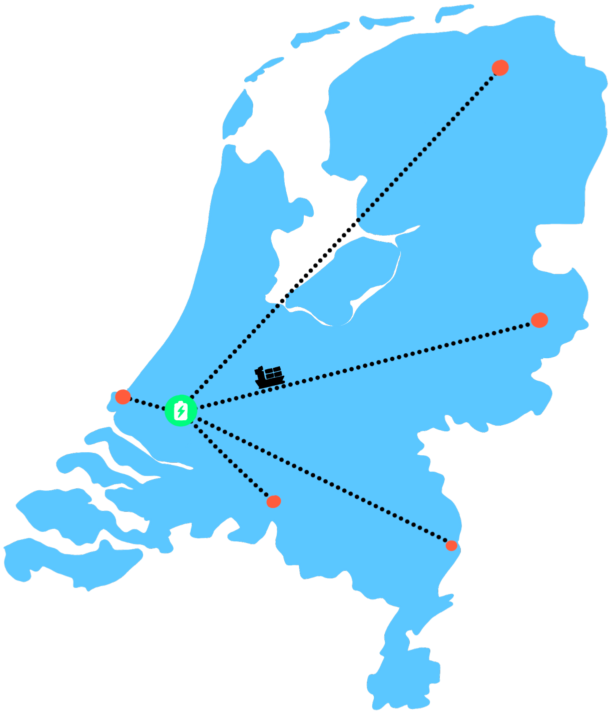


Figure 2: Schematic view of a charging port with connected routes on which the electrified vessels could move through the inland waterways of the Netherlands

2.2.1 Model Assumptions

To effectively model the introduced waterway system, it is necessary to simplify the reality. In this regard, a set of assumptions is formulated, which are reasonable within the context of this research. First, we give a list of the assumptions and thereafter it is explained why these assumptions are reasonable in the context of this research. The following assumptions are made:

Assumption 2.1 (Single Port)**.** There is only a single port that is equipped with all the charging stations, which all the ships travel through.

Assumption 2.2 (Set Routes)**.** All vessels travel a predetermined route, with the beginning and end of the routes at the charging port.

Assumption 2.3 (Reducing to Event Time-Space)**.** The full timeline can be reduced to a discrete set of event times.

Assumption 2.4 (Asynchronous Event Times)**.** No two events happen simultaneously, they could occur arbitrarily close to each other.

Assumption 2.5 (No Harboring Times)**.** The harboring practices occur instantaneously. These activities include moving cargo and batteries as well as connecting and disconnecting batteries to charging stations.

Assumption 2.6 (No Waiting Times)**.** The vessels do not wait for the batteries to recharge, a schedule must have a charged battery at the time of arrival for all ships.

Assumption 2.7 (Docking Space)**.** There is no constraint in harboring space at the harbor for the vessels.

Assumption 2.8 (Battery Equivalence)**.** All batteries are equal and no battery degradation occurs in the scheduling period chosen for this research.

Assumption 2.9 (Charging Station Equivalence)**.** Charging stations are equal and perform the same throughout the day.

Assumption 2.10 (Single Battery)**.** All vessel journeys and grid balancing stints use a singular battery.

Assumption 2.11 (Fixed Grid Balancing Times)**.** The start and end times of the grid balancing stints are known.

Assumption 2.12 (Fixed Grid Balancing Revenue)**.** The revenue of taking all the grid balancing stints is known.

Assumption 2.13 (Capturing the Uncertainty)**.** All system uncertainty can be captured in two parameters: energy consumption uncertainty and arrival time uncertainty.

Assumption 2.14 (Uncertainty is Estimable)**.** The probability distribution of both uncertainties is known or can be approximated.

Assumption 2.1 refers to the scope of this research being limited to a system with a singular charging port. Assumption 2.2 ensures that the vessels are decently predictable and the distribution of their arrival times can be estimated on historical data. The system this research is interested in transitioning is defined by these two assumptions. In practice, such a system is retrieved from the full Dutch waterway system as seen in Figure 1. Extract a number of vessels that sail a predetermined back-and-forth route with a common loading port as seen in Figure 2. From this simplified network, we are only interested in the activities at the port where the charging stations can be placed. Figure 3 shows an example of how the port could be located. The figure also shows the locations where a battery container can be stationed: on a charging station, in storage, or on a vessel.

Assumption 2.3 reduces the continuous timeline to a viable event time-space on which the system can be modeled. Due to the reduction from a continuous timeline to a discrete event time-space, it is reasonable to assume Assumption 2.4, since the probability that two events occur exactly at the same time in a continuous timeline equals zero.

Harboring times are included in the voyage time for this research. Therefore, Assumption 2.5 is introduced to have a single event where batteries are switched. A charged battery schedule that is produced by operators needs a battery to be ready at the moment that this event occurs otherwise the schedule is deemed not suitable. This is ensured by Assumption 2.6.

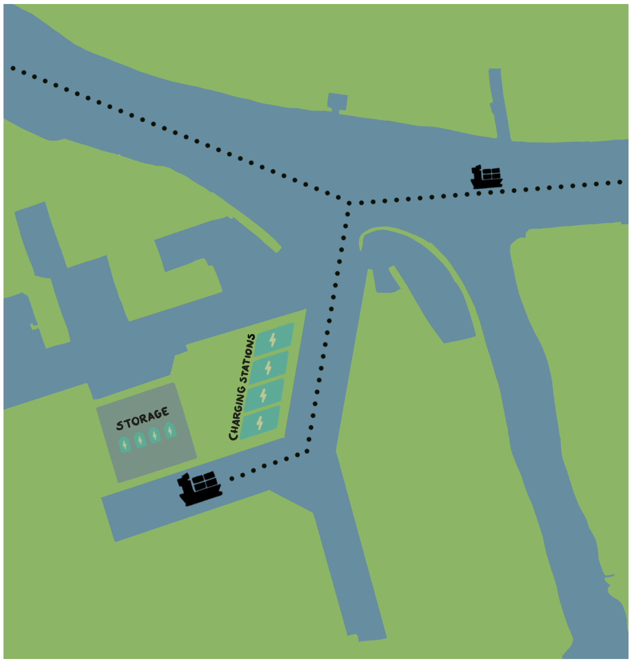


Figure 3: Schematic view of a fictional charging port where the positional nodes of the batteries are shown. (Note that this figure is not to scale.)

The spatial limitations of the system are excluded from the scope. Assumption 2.7 guarantees that these limitations do not impact the scheduling process for the charging port.

Assumption 2.8 is reasonable, because the battery is viewed as an input for this research. Hence, the batteries are assumed to be fixed and equal for this problem. The same argument holds for the charging stations in Assumption 2.9.

Assumption 2.10 ensures that the charging levels at returning the battery can be determined. Given degradation or multiple batteries, the exact electric charge left in the batteries can be very hard to know.

Grid balancing or trading opportunities are also considered to be an input in this research. Assumption 2.11 establishes the specific time frame in which the balancing occurs. These times are agreed upon far in advance.

As mentioned in the introduction, two uncertainties are considered in this research: energy consumption uncertainty and arrival time uncertainty. A few assumptions about the uncertainties of the system are made. As mentioned in the introduction, supply and demand, mass transportation, and the weather are not directly included in the scope of this research. However, these uncertainties indirectly influence the uncertainties that are included in the scope. Assumption 2.13 assumes that they are fully captured by the uncertainties included, such that the model is a full representation of the problem.

To be able to model the uncertainties included in the scope, Assumption 2.14 ensures that their probability distribution can be determined. Then, a model can be built that includes the uncertainties. Given all these assumptions, a mathematical formulation that serves as a foundation for the model can be made for an electrified inland waterway system.

2.2.2 Mathematical Formulation

The waterway system under the previously stated assumptions can be described as a event time-space graph $\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{A}^t)$, where the set of nodes $\mathcal{N}$ consists of all the locations a battery can be stationed. Therefore, $\mathcal{N}$ is divided into three subsets: ships, storage, and charging station stations. The set $\mathcal{T}$ is the set of the discrete time-event space indexed by t. There are three types of events in the model:

1. the moment a ship arrives at the charging port;
2. the moment a grid balancing contract starts;
3. the moment a grid balancing contract ends.

The first two events are seen as the start of a battery *task*. The tasks end if the same ship returns or if the same grid balancing contracted is ended, then the battery returns to the charging harbor system. The choice to include the end of a grid balancing task and not the end of a vessel task is made due to the optionality of grid balancing. Grid balancing stint can be viewed as an arbitrary revenue option and does not continuously in need of a battery.

The following decisions are made throughout the system:

1. At the start of a task, ship arrives or grid balancing stint, the decision is taken to allocate a battery to a vessel or optionally to grid balancing;
2. Between time events, a decision is taken to allocate charging time to batteries. This is facilitated by the instant swapping of batteries (see Assumption 2.5), which allows multiple batteries to be attached to the available charging stations within the time delta.

The set of arcs $\mathcal{A}^t = (n_1, n_2)^t$ represents the options for moving a battery at time t. The possibilities are limited to moving a battery to the just-arrived ship, to storage, or to a charging station. The available arcs are highly dependent on the current time. For example, a battery can only be placed on a ship if that ship is harbored at the current port.

Figure 4 shows the theoretical time-event space of a single harbor system. This timeline represents the information a harbor planner has in real-time. A certain distribution gives the arrival times of ships over a time period. This results in a confidence interval for the time in which the ship will arrive at the port. is often graphically represented as a box as seen in Figure 4.

Figure 5 shows the time-event space in hindsight. In hindsight, a perfect solution can be calculated. Planning strategies can be compared to this ideal solution for the performance of the decisions. This timeline is equivalent to the time event space in a deterministic version of the model.

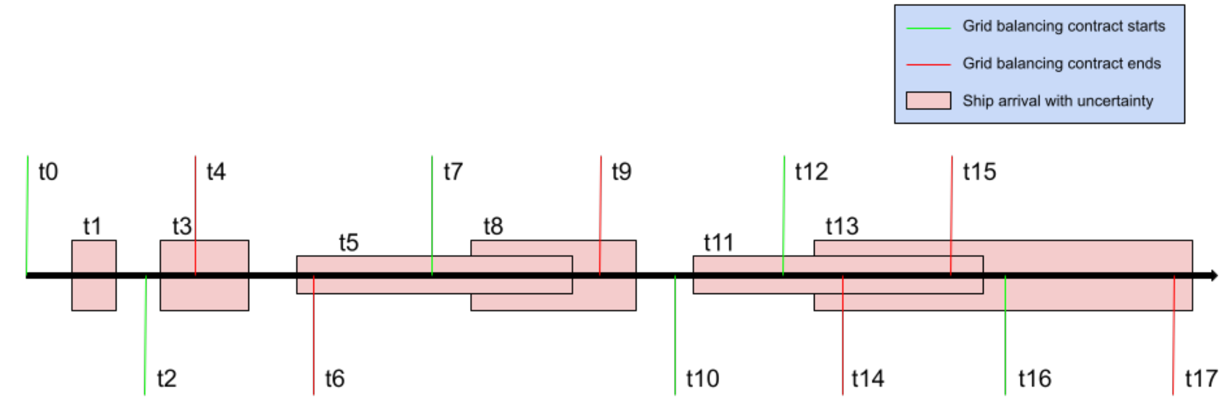

Grid balancing contract starts
Grid balancing contract ends
Ship arrival with uncertainty
t0
t4
t7
t9
t12
t15
t1
t3
t8
t13
t5
t11
t2
t6
t10
t14
t16
t17

Figure 4: The theoretical timeline of a port operator that makes the allocation decisions of the batteries. The arrival times of the ships are known to an extent. The further a vessel is from arriving at the port the more uncertain the arrival time is for the port operator. The grid balancing times are known beforehand and are therefore exactly represented. The length of the timeline is called the event horizon, this is the event time-space that includes all the events for which an operator plans.

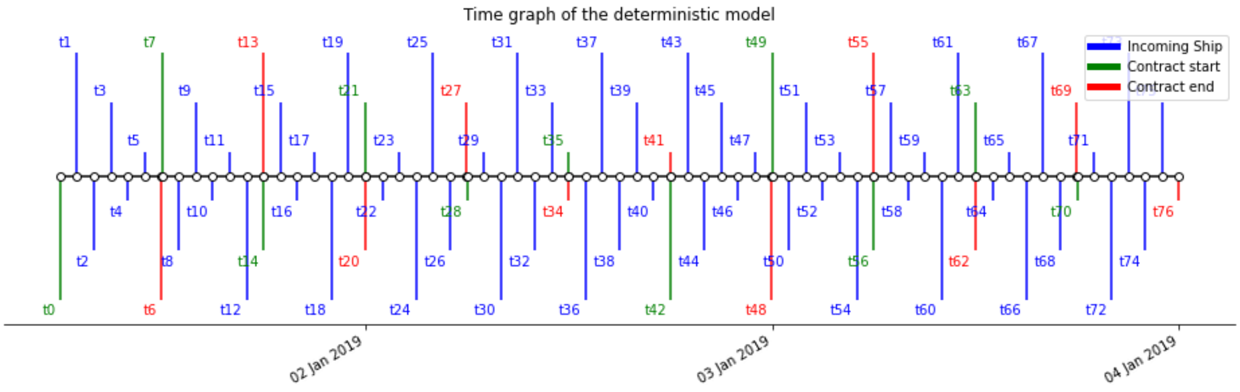


Figure 5: A realization of a three-day timeline at the charging harbor. This is a scenario that is only known in retrospect. It corresponds to the deterministic version of the timeline seen in Figure 4. This is a stylized case where the arrival times are exactly an hour apart. A timeline can be created by Monte Carlo sampling from the time uncertainty boxes seen in Figure 4.

2.3 Sets, Parameters, and Variables

In this section, we present the model and its attributes that form the basis of the mathematical problem formulation. First, the relevant sets associated with the problem are defined. Next, the known parameters and uncertain parameters pertaining to these sets are provided. Finally, we outline the decision variables that can be modified by the port operators.

2.3.1 Sets

The problem's main objects in the system are described in the following sets:

Table 1: Sets

Sets	**Indices**	**Descriptions**
$\mathcal{S}$	s	Ships
$\mathcal{G}$	g	Grid balancing spots
$\mathcal{B}$	b	Batteries
$\mathcal{T}$	t, t_1, t_2	Event time-space
$\mathcal{T}_a$	t_a	The time events where a ship arrives (t_s is the subset of the arrival times of vessel s)
$\mathcal{T}_g$	t_g	The time events where a grid balancing spot starts

Note that $\mathcal{T}_a$ and $\mathcal{T}_g$, by Assumption 2.4 are disjoint sets that make up the event space $\mathcal{T}$, i.e. $\mathcal{T}_a \cup \mathcal{T}_g = \mathcal{T}$ with $\mathcal{T}_a \cap \mathcal{T}_g = \emptyset$. Furthermore, the continuous timeline is discretized with a finite number of events ($\#\mathcal{T}$), then it is reasonable to assume that no two events happen exactly at the same time because the probability of two simultaneous events at a continuous timeline equals zero.

2.3.2 Known Parameters

The attributes of the system that are known before a schedule is made and are included in the research are:

Table 2: Parameters without uncertainty

Parameters	**Descriptions**
$t_{\min}$	A virtual starting point at the very start of the timeline
$t_{\max}$	A virtual ending point at the very end of the timeline
C_t^-	The minimum charge for a battery allocated at time t, greater than E_t (see Table 3)
C_t^+	The maximum charge for a battery allocated at time t, typically set to one
$[C_t^-, C_t^+]$	The interval of charge level that a battery must contain to be allocated at time event t
T_g	The timing of time-event t_g
d	The charging speed if a battery is connected to the grid for charging
e	The self-discharge rate
S_b	The starting position of battery b
c_b^0	The starting charge level of battery b
R_{t_g}	The revenue of grid balancing spot t_g

2.3.3 Parameters With Uncertainty

Not all parameters are known at the time of scheduling. Some parameters are uncertain. There are two types of uncertainties included in this research:

- uncertainty in the percentage of energy that is used to complete a voyage,
- uncertainty in the arrival times of the vessels, i.e. the timing of an arrival event in the event time-space (t_a).

The following parameters are dependent on the uncertainties stated above:

Table 3: Parameters with uncertainty

Parameters	Descriptions	Uncertainties
E_t	The energy consumption of the event starting on time t	Estimated normal distribution
T_a	The ETA prediction of arrival a with uncertainty	Distribution based historical data
Δ^t	The time difference between event t and event $t-1$	Dependent on T_a
A_{t_1,t_2}	The matrix that shows the relation of battery availability and time events (see Appendix B.1 for the full definition)	Dependent on the ordering of time events

2.3.4 Variables

The following variables are the attributes of the system that can be changed or chosen by the port operatives:

Table 4: Variables

Variables	Ranges	Descriptions
x_b^t	$\{0,1\}$	The decision variable that assigns a battery b at time t
c_b^t	$[0,1]$	The charge level of the battery b at time t
w_b^t	$[0,1]$	The decision variable that shows the percentage of time battery b is charging time frame $[t-1,\, t]$
y_b	$\{0,1\}$	The variable that indicates whether or not battery b is used somewhere in the schedule
B	$\mathbb{N}$	The number of batteries used.
M	$\mathbb{N}$	The number of charging stations used.

The triplet of variables (**x**,**c**,**w**) will be revered to as a *schedule*. Furthermore, the pair (B, M) will be called the *infrastructure* of the schedule.

2.4 Constraints

The sets of constraints are built up systemically. First, we start the set of constraints dependent on only the primary decision variable x_b^t that assigns batteries to either vessels or balancing spots. Then, we move up to the constraints on the charge level variable c_b^t; the charging w_b^t; the starting conditions; and lastly the infrastructural variables y_b and M.

2.4.1 Battery Allocation Constraints

Three constraints relating solely to the primary decision variable are needed. Firstly, we assume that all ships need a single battery to do their route. Therefore,

$$\sum_b x_b^{t_a} = 1, \quad \forall t_a \tag{1}$$

ensures that a single battery is used at every ship's arrival.

Note that this assumes that a single battery is used per ship (Assumption 2.10). This is not necessarily true in the real world. In that case, this constraint should be set to a parameter that denotes the batteries necessary for this vessel.

Secondly, we assume that a single battery is necessary for every grid balancing spot. Therefore,

$$\sum_b x_b^{t_g} \leq 1, \quad \forall t_g \tag{2}$$

ensures that at most a single battery is used for a grid balancing spot.

Note that this assumes that a single battery is used per balancing spot (Assumption 2.10). To be able to add trading opportunities that require multiple batteries an adjustment in the model is required because the R_t does not account for multiple batteries.

Lastly, batteries cannot be used if they are still in use somewhere else. Therefore,

$$\sum_t x_b^t A_{t_1,t} \leq 1, \quad \forall b, t_1 \tag{3}$$

ensures that if a battery b is assigned at time t, it is not available for the following few assignments until it has returned at the first zero in A_{t_1,t_2} at row t.

2.4.2 Charge Level Constraints

We introduce two constraints related to the charge level c_b^t given the x_b^t. We assume that vessels and energy companies agree to a contract that includes a minimum charge level required. Therefore,

$$c_b^t \geq C_t^- \cdot x_b^t, \quad \forall b, t. \tag{4}$$

This constraint ensures a battery chosen at time t has enough charge for the ship or balancing spot.

There also exists a maximum for the charging level of the allocated battery:

$$c_b^t \leq C_t^+ \cdot x_b^t + (1 - x_b^t), \quad \forall b, t \tag{5}$$

the latter part again ensures that this constraint only holds if battery b is assigned at time t. Normally the C_t^+ is equal to 1 for a vessel arrival time ($t \in \mathcal{T}_a$) because there is no disadvantage to having a battery with too much charge on a vessel. However, for grid balancing there may be an upper limit to the charging level of the battery because they might need to store more energy in the battery that is lent for grid balancing.

Also, batteries are used if they are assigned by x_b^t, this results in a shift in charge level that takes effect directly after the battery is assigned. Therefore,

$$c_b^{t+1} \leq c_b^t - E_t x_b^t - e\Delta^{t+1} + (1 - x_b^t), \quad \forall b, \forall t \in \mathcal{T}/\{t_{\max}\} \tag{6}$$

this ensures that the vessel trip or balancing stint consumes the amount that is required from the allocated battery (E_t). The term $(1 - x_b^t)$ of the constraint ensures that the constraint only holds if the battery is chosen at time t, otherwise the left-hand side of the constraint will be set higher than the maximum range of c_b^{t+1}, which is equal to one. The term $e\Delta^{t+1}$ is the standard discharge of the battery.

Note that the system can drop energy at will, due to the inequality. However, there is no

2.4.3 Charging Decision Constraints

We assume that the charging attributes of the pair-wise grouping batteries and charging stations are the same (2.8, 2.9). The most a battery can be charged at a station is the charging speed d multiplied by the time the battery is charging. The same statement holds for self-discharging, the self-discharge speed e multiplied by the time a battery is not charged.

$$c_b^{t+1} \leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}), \quad \forall b, \forall t \in \mathcal{T}/\{t_{\max}\}, \tag{7}$$

this gives batteries the option to be charged. The higher w_b^t is set the more time the battery b is spent charging in the time frame $(t, t+1)$. Note that a battery on a vessel also discharges at the rate of e.

Also, a battery cannot be charged if it is used on a ship or if it is reserved for grid balancing. Therefore,

$$x_b^{t_1} \sum_t (A_{t_1,t}) \leq \sum_t ((1 - w_b^t) A_{t_1,t}), \quad \forall b, t_1, \tag{8}$$

this forces w_b^t to be zero if the battery b is away and gives the possibility to charge if the battery is available.

2.4.4 Starting Conditions

The starting conditions for the batteries are also specified in the model. Two conditions are defined in the model, a starting position and a starting charge level. Therefore, we have the following constraints:

$$x_b^t = 0 \text{ and } w_b^t = 0, \quad b, t | t \in t_{\min} \leq t < t_{S_b}, \tag{9}$$

where t_{S_b} is equal to the first arrival time of vessel S_b and equal to $t_{\min}$ if the starting position is the charging port. This constraint prevents the usage of the batteries where the starting position is on a vessel.

$$c_b^{t_{\min}} <= c_b^0, \quad \forall b, \tag{10}$$

this ensures that optimization starts with almost the predefined starting charge levels.

2.4.5 Infrastructural Usage Constraints

The infrastructure also set limitations on the system. The system needs to keep track of how many batteries are in use and the batteries charging may never exceed the number of charging stations present. Therefore, we have the following constraints:

$$N y_b \geq \sum_t x_b^t, \quad \forall b, t \tag{11}$$

where B is large enough. This ensures y_b is set to one if b is used somewhere in the system.

$$\sum_{b}(w_b^t + \sum_{t_g} x_b^{t_g} A_{t_g,t}) \leq M, \quad \forall t, \tag{12}$$

this ensures that the amount of charging stations M is larger than the number of charging stations used.

2.5 Objective Functions

In this research, we are interested in two questions:

1. How many battery containers and charging stations are needed to fully transform a waterway system?
2. How do you optimally plan the battery containers and charging stations to make as much optimal profit from grid balancing?

These two questions lead to two distinct problems with different objective functions. The first question minimizes the number of batteries and charging stations for which a feasible schedule exists. This problem is a bi-objective minimization of both the number of batteries and charging stations. The second question maximizes the revenue gained from accepting the grid balancing stints.

2.5.1 Feasibility objective functions

The first goal of this research is to determine the infrastructure necessary to realize the transition of an inland waterway system to make use of only electric-powered vessels. The infrastructure consists of batteries and charging stations (B, M). Therefore, two objectives determine the minimal value for respectably both. The first objective function minimizes the number of batteries used in the schedule.

$$\begin{gathered} \text{minimize } \sum_{b} y_b \\ \text{subject to constraints (1, 3 - 12)} \end{gathered} \tag{13}$$

where $B = \sum_b y_b$. The second objective function minimizes the number of charging stations used in the schedule.

$$\begin{gathered} \text{minimize } M \\ \text{subject to constraints (1, 3 - 12).} \end{gathered} \tag{14}$$

This problem will be referred to as the *Feasibility Problem* (*FP*), the full mathematical description can be found in Appendix B.3. Objectives (13) and (14) are conflicting in the sense that with more charging stations, there is the possibility that fewer batteries are necessary. There is a trade-off between these two objectives. Therefore, if the objectives are combined we have a multi-objective optimization with conflicting objective functions. The bi-objective optimization has a set of solutions that are Pareto optimal, this is defined and explained in chapter 4. Note that there is no incentive to do any grid balancing. In the context of *FP*, it is safe to assume no optional grid balancing spots will be used in these combined objectives.

2.5.2 Revenue objective function

In the system, shipping always has priority over grid balancing. However, the decision to lease the batteries to the network operators should be taken as often as possible for *optimal revenue*. The objective of the optimization will be to maximize the sum over all grid spots. Therefore the optimization problem becomes:

$$\begin{aligned} &\text{maximize } \sum_{t_g} \sum_{b} R_{t_g} x_b^{t_g} \\ &\text{subject to constraints } (1 \text{ - } 9, 12) \, . \end{aligned} \tag{15}$$

This problem will be referred to as the *Revenue Problem* (RP), the full mathematical description can be found in Appendix B.4.

3 Complexity Analysis

In this chapter, we will discuss the complexity features and approximation methods of the problem described in Chapter 2. The complexity classes are all defined on the set of decision problems. In the complexity section, we will prove that the decision version of *FP* is NP-Hard and that *FP* without charging is strongly $\mathcal{NP}$-complete. In the approximation section, we discuss the best bounds and approximations that exist in the literature on comparable decision problems. The definitions of the complexity classes and the problems used for the reduction can be found in Appendix A.

3.1 Complexity

This section discusses the complexity classes of the decision versions of both problems. First, the Feasibility Problem is analyzed by comparing it to the Bin Packing Problem. Then, the Revenue Problem is compared in complexity to the Feasibility Problem.

3.1.1 Complexity of the Feasibility Problem

To investigate the complexity of our *feasibility problem* (*FP*), we rewrite the problem as a decision problem (FP-dec). Suppose you are given a certain amount of batteries and charging stations, a yes-instance of *FP-dec* denotes the existence of a schedule (**x,c,w**) for the batteries over time such that all vessels can be facilitated with a battery. A no-instance of *FP-dec* denotes the impossibility of such a schedule.

Definition 3.1 (FP-dec)**.** For a given B, M the decision version of the FP problem can be formulated by the question: " Given B batteries ($y_b = 1$ for a set of b with cardinality B) and M charging stations, does a schedule exist such that all the ships are provided with a charged battery at the time of arrival?"

$$\text{FP-dec}(\mathcal{T}, C_t, d, e, B, M) = \left\{ \begin{array}{c} \exists \mathbf{x} \text{ with corresponding } \mathbf{c,w} \\ \text{such that constraints (1, 3 - 12) hold} \\ \text{given that } \sum_b y_b = B \text{ and } M \text{ fixed} \end{array} \right\} \tag{16}$$

This can be written out to be the following system:

Find a schedule($\mathbf{x,w,c}$) such that:

$$\begin{aligned}
\sum_b y_b &= B \\
\sum_b x_b^t &= 1 \\
\sum_t x_b^t A_{t_1,t} &\leq 1 \\
c_b^t &\geq C_t \cdot x_b^t \\
c_b^{t+1} &\leq c_b^t - E_t x_b^t + (1 - x_b^t) \\
c_b^{t+1} &\leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) \\
x_b^{t_1} \sum_t (A_{t_1,t}) &\leq \sum_t ((1 - w_b^t) A_{t_1,t}) \\
N y_b &\geq \sum_t x_b^t \\
M &\geq \sum_b w_b^t \\
x_b^t &\in \{0,1\} \\
c_b^t, w_b^t &\in [0,1]
\end{aligned} \tag{17}$$

Firstly, we show that *FP-dec* is in $\mathcal{NP}$, by showing it is verifiable in polynomial time. Secondly, we prove that *FP-dec* is strongly $\mathcal{NP}$-Hard. We use a polynomial time reduction to the Bin Packing Problem (*BBP*) and its polynomial time reduction to the 3-partition problem.

Theorem 3.2. *FP-dec is in* $\mathcal{NP}$

Proof. Suppose we have a yes-instance of *FP-dec*. Then, a schedule ($\mathbf{x,c,w}$) with B batteries and M charging stations exists. Since all constraints in *FP-Dec* are linear and the number of constraints is linear to the input size. The schedule ($\mathbf{x,c,w}$) extracted from the yes-instance can be filled in the constraints and objective function in polynomial time, to verify the objective. Therefore, *FP-dec* is in $\mathcal{NP}$. □

Before delving into proving that *FP-dec* is strongly $\mathcal{NP}$-Hard, two preliminary lemmas are defined and proven. These lemmas form the foundation of the proof that *FP-dec* is $\mathcal{NP}$-Hard.

Given an instance of the *BPP* as defined in A.6. We are given a set of items $\mathcal{S} = 1, \cdots, n$ indexed by i that has size $s_i \in (0, 1]$ and a set $\mathcal{K} = 1, \cdots, n$ of knapsacks (bins) indexed by k with capacity one. Then a possible mathematical formulation of *BBP* formulated by Martello and Toth [1990] is:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{k=1}^{n} y_k & \\
\text{subject to} \quad & \sum_{i=1}^{n} s_i x_{ki} \leq y_k, & \forall k. \\
& \sum_{k=1}^{n} x_{ki} = 1, & \forall i. \\
& y_k \in \{0,1\}, & \forall k. \\
& x_{ki} \in \{0,1\}, & \forall k, i.
\end{aligned}
$$

Definition 3.3 (*BBP-Dec*)**.** The decision version of *BBP* can be formulated by the following question. "Does a packing exist such that all items $i \in I$ with sizes $s(i)$ are contained in the K bins?". A yes-instance denotes the existence of such a packing, and a no-instance the reverse.

A mathematical formulation for *BBP-Dec* can be constructed from the Martello and Toth [1990] formulation. This results in the following linear system for *BBP-Dec*:

$$
\begin{aligned}
& \text{Find } (\mathbf{y,x}) \text{ such that:} \\
& \sum_{k} y_k = K \\
& \sum_{i=1}^{n} s_i x_{ki} \leq y_k, \forall k. \\
& \sum_{k=1}^{n} x_{ki} = 1, \forall i. \\
& y_k \in \{0,1\}, \forall k. \\
& x_{ki} \in \{0,1\}, \forall k, i.
\end{aligned}
\tag{18}
$$

Lemma 3.4. *FP-Dec is polynomial-time reducible to BPP-Dec*

Proof. The following proof shows that *BBP-Dec* is a special case of the *FP-Dec*, with no charging or discharging ($d, e = 0, w_b^t = 0$) and a singular ship (matrix A is the identity matrix). Also, the feasible charge range to allocate a battery at time t, $[C_t^-, C_t^+]$, will be set to E_t.

Consider the system given in equation 17. Now we will set all the charge and discharge parameters to 0. This leaves the following system of equations:

$$\begin{aligned}\sum_b y_b &= B\\ \sum_b x_b^t &= 1\\ \sum_t x_b^t A_{t_1,t} &\leq 1\\ c_b^t &\geq C_t \cdot x_b^t\\ c_b^{t+1} &\leq c_b^t - E_t x_b^t + (1 - x_b^t)\\ c_b^{t+1} &\leq c_b^t\\ y_b, x_b^t &\in \{0,1\}\\ c_b^t &\in [0,1]\end{aligned}$$

Now, we assume that all batteries start with a full charge. This problem above states that at every time step a certain amount of power should be taken from a battery to facilitate the ship's next voyage. This can be reformulated by stating that the E_t is the power of the task t and that this power should be removed from a battery b using a decision variable x_b^t. The formulation is equivalent to the following representation with the bookkeeping variable c_b^t eliminated. This results in the following system of equations:

$$\begin{aligned}\sum_b y_b &= B\\ \sum_{t=1}^{n} E_t x_{bt} &\leq y_b\\ \sum_{b=1}^{n} x_{bt} &= 1\\ y_b, x_b^t &\in \{0,1\}\end{aligned}$$

The formulation is the same as the *BPP-dec* stated in Equation (18), where the set of tasks $\mathcal{T}$ is equivalent to the set of items $\mathcal{S}$ with their respectable sizes E_t equivalent to s_i. Also the set of batteries $\mathcal{B}$ is equivalent to the set of bins $\mathcal{K}$. Thus, if we find a yes-instance of the *FP-Dec* problem we also find a yes-instance for the arbitrary *BPP-Dec* problem. All the reductions are made in polynomial time. Therefore, every instance of *FP-Dec* is reducible to an arbitrary instance of *BPP-Dec* in polynomial time. If there exists a polynomial-time algorithm that solves *FP-Dec*, the algorithm would also solve *BBP-Dec*. □

Lemma 3.5. *The BBP-Dec is strongly $\mathcal{NP}$-Hard*

Proof. This is proven by the reduction of *BBP-Dec* to the 3-Partition Problem (*3PP*). Given an instance $\hat{I}$ of the *3PP* as defined in A.5. Garey and Johnson [1979] define the instance of can be written as follows: Given a sets of items I with $3K$ elements and a $\beta \in \mathbb{N}$ and a mapping $w : I \to \mathbb{N}$ such that $\beta/4 < w(i) < \beta/2$ and $\sum_{i \in I} w(i) = k\beta$. The decision question of the *3PP*: "Does a partition exist of tuples of size three such that all sums of the sets equal β?".

Define $s(i) = \frac{w(i)}{\beta}$. Note that $1/4 < s(i) < 1/2$, by the bounds defined on $w(i)$. Consider the *BBP-Dec* instance that consists of K bins and items i with their respective sizes set to $s(i)$, with the decision question: "Does there exist a packing such that all items are packed within K Bins?". This instance of *BBP-Dec* is equivalent to the $\hat{I}$ instance of *3PP*, because exactly 3 items must be packed without any room left in all K bins to get a yes-instance. Thus, if a solution for this instance of *BBP-Dec* is found, there follows a solution for $\hat{I}$ instantly, i.e. within polynomial-time. Therefore, *BBP-Dec* is reducible to *3PP* in polynomial time.

Now, it is sufficient to prove that *3PP* is strongly $\mathcal{NP}$-Hard. That proof can be found in Garey and Johnson [1979]. *3PP* is described to be the 'basic' $\mathcal{NP}$-complete problem in the strong sense. □

Using Theorem 3.2 and Lemmas 3.4 and 3.5, we can prove the following theorem:

Theorem 3.6. *FP-Dec is strongly NP-complete*

Proof. By Definition A.11, a decision problem is said to be strongly $\mathcal{NP}$-complete if it is contained in $\mathcal{NP}$ and is strongly $\mathcal{NP}$-Hard. The first part of the definition is established by Theorem 3.2. Using Lemma 3.4, we can conclude that *FP* is polynomial-time reducible to *BPP-dec*. Because of the transitive property of polynomial-time reducibility and Lemma 3.5, we know that *FP-Dec* is polynomial-time reducible to 3PP and therefore strongly $\mathcal{NP}$-Hard. □

3.1.2 Complexity of the Revenue Problem

Similar to the previous section, we can prove that the decision version of *RP* is also strongly $\mathcal{NP}-$Complete.

The decision version of the RP includes an extra parameter R that represents the minimum revenue to make the problem a 'yes'-instance. Therefore, the question of the *RP-dec* becomes: "Does a schedule exist such that the revenue is greater than R?".

Definition 3.7 (RP-Dec)**.**

$$\text{RP-Dec}(\mathcal{T}, C_t, d, e, B, M, R) = \left\{ \begin{array}{c} \exists \mathbf{x} \text{ with corresponding } \mathbf{c},\mathbf{w} \\ \text{such that constraints (1 - 12) hold given that} \\ \sum_b y_b = B,\ M \text{ fixed and } \sum_{t_g} \sum_b R_{t_g} x_b^{t_g} \geq R \end{array} \right\} \qquad (19)$$

The procedure to prove the complexity is identical as done for *FP-Dec*. Firstly, we show that *RP-dec* is in $\mathcal{NP}$, by showing a yes-instance is verifiable in polynomial time. Secondly, we prove that *RP-dec* is strongly $\mathcal{NP}$-Complete. We use a polynomial time reduction to *FP-dec*.

Theorem 3.8. *RP-dec is in $\mathcal{NP}$*

Proof. Suppose we have a yes-instance of *RP-dec*. Then, there exists a schedule (**x**,**c**,**w**) with B batteries and M charging stations with a revenue greater than R. Since all constraints in *RP-Dec* are linear, we can fill the schedule (**x**,**c**,**w**) in the provided by the yes-instance into the system above in polynomial time. Therefore, *RP-dec* is in $\mathcal{NP}$. □

RP-Dec is a direct extension of *FP-Dec*. The extended part consists of the fact that the schedule does not only need feasible with K batteries and M charging stations but also has to generate a certain amount of revenue R. Therefore, a direct reduction to *FP-Dec* exists and we can prove that *RP-Dec* is also strongly $\mathcal{NP}$-Complete.

Theorem 3.9. *RP-Dec is strongly $\mathcal{NP}$-complete.*

Proof. By Definition A.11, a decision problem is said to be strongly $\mathcal{NP}$-complete if it is contained in $\mathcal{NP}$ and is strongly $\mathcal{NP}$-Hard. The first condition is satisfied by Theorem 3.8.

Given an instance $\hat{I}$ of *FP-Dec* as stated in Definition 3.1. Consider the instance I of *RP-Dec* where everything is the same as in $\hat{I}$ with additionally $R_{t_g} = 0$ for all t_g and $R = 0$. Then, the constraint $\sum_{t_g} \sum_b R_{t_g} x_b^{t_g} \geq R$ always holds and the remainder of instant I equals $\hat{I}$. If there would exist a polynomial-time algorithm that solves *RP-Dec* it would also solve *FP-Dec*.

Since *FP-Dec* is strongly $\mathcal{NP}$-Hard by 3.6, *RP-Dec* is also strongly $\mathcal{NP}$-Hard. □

3.2 Approximation

After establishing that both problems are $\mathcal{NP}$-Complete, it can be concluded that no polynomial time algorithms exist to find an optimal solution for all instances. However, a follow-up question remains: what polynomial algorithms do exist that find nearly optimal solutions to the two optimization problems? Near-optimal problems are called approximation algorithms. The approximation algorithms for our problem are in the form of a constant-factor approximation scheme.

Definition 3.10 (Constant-factor Approximation Scheme (CFAS))**.** CFAS is a polynomial-time algorithm with a factor $\gamma > 0$ that solves a problem within a factor of $\gamma \geq 1$ from its optimal for all instances of the problem. If for a problem P there exists a polynomial time algorithm within $\gamma \cdot OPT(I)$ for all instances I, we say that P has CFAS of γ.

This section provides lower bounds and an approximation method for the *FP* that serve as upper bounds to the performance of approximation algorithms for this problem.

3.2.1 Lower bound of approximation algorithms of the Feasibility Problem

The Feasibility Problem as seen in the previous has a close relation to bin backing. Therefore, we look at approximation algorithms for *BBP*. Firstly we look at the lower bounds found for *BBP*, then an algorithm is given that serves as an upper bound for the best CFAS.

Theorem 3.11. *BBP has no CFAS smaller than 3/2 For any $\epsilon > 0$, there is no approximation algorithm having a guarantee of $(3/2 - \epsilon) \cdot OPT(I)$ for all instances I of the bin packing problem, assuming $\mathcal{P} \neq \mathcal{NP}$.*

Proof. This proof is inspired by Vazirani [2003]. Consider an instance I_0 of the Partition problem A.4, with $S = s_1, \cdots, s_n$. Assume that there exists a polynomial-time approximation algorithm A for *BBP* such that $A(I) < 3/2OPT(I)$ for all instances I of bin packing. Without loss of generality, we can assume that $A(I), OPT(I) > 1$ for all I otherwise there is nothing to optimize. It holds for all I, thus it holds for $\hat{I}$ defined by the capacity of the bins (c) is equal to $\sum_i s_i/2$. Now consider two types of answers the algorithm can produce. First $A(\hat{I}) = 2$, then the I_0 is a yes-instance because a partition over two sets (bin 1 and bin 2) can be made. On the other hand if $A(\hat{I}) \geq 3$, then we know that $3 < 3/2 \cdot OPT(\hat{I})$, i.e., $OPT(\hat{I}) > 2$, then the I_0 is a no-instance because the set S cannot be partitioned. It follows that we can solve the Partition problem in polynomial-time with algorithm A. This is a contradiction if $\mathcal{P} \neq \mathcal{NP}$ because the Partition problem is a $\mathcal{NP}$ problem, proven by Garey and Johnson [1979].

□

Theorem 3.12. *FP has no CFAS lower than 3/2 for the number of batteries objective For any $\epsilon > 0$, there is no approximation algorithm having a guarantee of $(3/2 - \epsilon) \cdot OPTB(I)$ for all instances I of the FP, assuming $\mathcal{P} \neq \mathcal{NP}$. Here $OPTB(I)$ is the optimal number of batteries given a number of charging stations for instance I.*

Proof. Assume that there exists a polynomial-time approximation algorithm A for *FP* such that $A(I) < 3/2 \cdot OPTB(I)$ for all instances I of *FP*. The same reduction steps can be done as in the proof of Lemma 3.4 to show that algorithm A would also solve all instances J of *BBP* with $A(J) < 3/2 \cdot OPT(J)$. Theorem 3.11 states that this is not possible if $\mathcal{P} \neq \mathcal{NP}$. □

3.2.2 First Fit Approximation algorithm for the Feasibility Problem

We construct a first-fit algorithm for *FP*. This algorithm always chooses the first in line to allocate and charge. The new batteries are stored at the end of the line. This is an extension of the first-fit algorithm used for *BBP* that can be found in Appendix A.14.

Algorithm 3.13 (First-fit Approximation algorithm *FP*)**.** *Given a system described in Section 2.*

```
1. Start with a singular battery $b$ on shore with a return time ($Time_b$) set to $T_0$, and $chargingtime_b$ set to $0$. The number of charging stations $M = 0$.
2. For all $t$ in $\mathcal{T}$ do:
    (a) If the charging level $c^t_{b_1}$ of the first battery $b_1$ is lower than $C^-_t$:
        i. If $(T_t - Time_{b_1} - chargingtime_{b_1}) * d > C^-_t - c^t_{b_1}$, add an extra charging station to the system and increase $c^t_b$ with $(T_t - Time_b - chargingtime_{b_1}) * d$
        ii. Else, add an extra battery to the system with a full charge and set the battery to the first battery.
    (b) Allocate the first battery $b_1$ with $c^{t+1}_{b_1} = c^t_{b_1} - Et$ to the arrived vessel and set the battery $b$ that comes in to be the last battery and set $Time_b$ to be the current time and the $chargingtime_b$ set to $0$.
    (c) Charge batteries from front to back $c^{t+1}_{b_i} = d \cdot \Delta_t + c^t_{b_i}$ for all $i \leq M$. Also, add the charging time to the first $i$ batteries until full then swap to the next battery left in the row. The charging time is trickled down to the following batteries.
3. Return $B$ to be the number of batteries and $M$ the number of charging stations.
```

The strategy of this approximation battery is ordering the batteries by charge level and matching them to their respective arrival times, the first battery is matched with the first arrival. In the event of a failure, i.e. no battery present with enough charge to facilitate the swap, a rapid troubleshooting calculation is performed to determine whether a new battery or a new charging station is added to the system.

4 Pareto Optimality

The Feasibility Problem described in Chapter 2 takes two objectives into account that have conflicting features. This conflict stems from the realization that adding more batteries to the system may result in using fewer charging stations. If the feasibility objectives (13, 14) are combined in a bi-objective optimization, the conflict likely results in that no global optimal exists for the problem. There is a trade-off in optimality within the two objectives. Therefore, we construct a set of solutions which are *Pareto optimal*. This set consists of locally optimal solutions of the *FP*. This chapter consists of two sections. The first section formally defines Pareto optimality for a multi-objective optimization problem. The second section describes two methods that construct the Pareto optimal set, the so-called Pareto front, namely the one-by-one optimization method and the weighted sum method.

4.1 Definitions

Pareto optimality only occurs when there are conflicting objective functions that should be simultaneously optimized. Therefore, this optimality only exists in a multi-objective optimization problem. This section is based on Coello et al. [2007].

Definition 4.1 (Multi-objective optimization problem (MOP))**.** The multi-objective problem of dimension d in the general form can be written as:

$$\begin{aligned}
&\text{minimize:}\\
&\qquad F(x) = [f_1(x), f_2(x), \cdots f_d(x)]\\
&\text{subject to:}\\
&\qquad g_j(x) = 0, j \in \{1, \cdots, n_1\}\\
&\qquad h_l(x) \leq 0, l \in \{1, \cdots, n_2\}\\
&\qquad x_i \in [l_i(x), u_i(x)], i \in \{1, \cdots, m\}
\end{aligned} \tag{20}$$

where x_i set of variables with lower limit l_i and upper limit u_i and where the functions $f_k, g_j, h_l : \mathbb{R}^m \to \mathbb{R}, \forall k, j, l$. The general multi-objective problem exerts a feasibility space with criterion space.

Definition 4.2 (Feasibility space)**.** The *feasibility space* is the set of feasible inputs of an optimization problem. Let the *feasibility space* of the multi-objective problem be X, i.e.

$$X = \{x | g_j(x) = 0 \text{ and } h_l \leq 0 \text{ and } x_i \in [l_i(x), u_i(x)]\}$$

Definition 4.3 (Feasibility criterion space)**.** The range of the feasible space in the objective function is called the feasibility criterion space. Let the *feasible criterion space* of the multi-objective problem be Z, i.e.

$$Z = \{F(x) | x \in X\}$$

On these feasibility spaces, the formal definition of Pareto optimality Pareto [1906] can be defined for the generalized MOP problem 4.1.

Definition 4.4 (Pareto optimal)**.** A point x^* is called Pareto optimal in the MOP problem 4.1 if and only if there does not exist another $x \in X$ such that $f_k(x) \leq f_k(x^*_k)$, for all $k = 1, 2, \cdots, d$ and there is at least one k such that $f_k(x) < f_k(x^*_k)$.

Definition 4.5 (Pareto Optimal Set)**.** The Pareto Optimal Set $\mathcal{P}^*$ of the MOP problem 4.1 is the set of all Pareto optimal solutions.

Definition 4.6 (Pareto front)**.** The Pareto front $\mathcal{PF}^*$ of the MOP problem 4.1 is the set of objective vectors of the Pareto Optimal Set, i.e.

$$\mathcal{PF}^* = \{u = F(x) | x \in \mathcal{P}^*\}. \tag{21}$$

4.2 Methods To Find Pareto Front

Coello et al. [2007] describe an extensive overview of various methods that are used to approximate the Pareto front. Two of these methods are considered in this thesis to find the Pareto front of the *FP*, namely one-by-one optimization and weighted sum. These methods are selected because they are easy to implement and suffice for our two-dimensional multi-objective optimization problem *FP*. Both their advantages and disadvantages and limitations are discussed, together with their application in the context of *FP*. As a result, the one-by-one method is identified as the preferred approach for the remainder of this research.

4.2.1 One-by-one Optimization Method

The one-by-one optimization is an iterative method that involves marginal optimization of all objectives. The main idea of this method is to optimize a single objective while keeping the other objectives fixed.

Firstly, the objectives are ordered in a sequence. Then, one by one, according to the sequence, all objectives are optimized fixing the previously optimized objectives to their optimal value. This method will always end in a Pareto optimal point because no objective function can be improved upon without worsening another. The resulting solution obtained from the method is highly dependent on the order of the sequence of objectives, a different order might end up in a different Pareto optimal solution. To construct a Pareto front, all the solutions of all the orderings of the objectives are combined.

The method finds all of the individually optimal solutions for all functions. A flaw of this method is that it misses trade-off solutions, for example, the solutions where no objective reaches their individual optimal value.

To incorporate trade-off solutions into the iterative method, an extension can be introduced. This extension relaxes the previous optimization criteria to explicitly examine if other optimizations can be improved without necessarily reaching their marginal optimal or previously discovered optimal solutions. This extension is particularly suitable for problems with a discrete solution space Z, as there is a finite set of options available to achieve a trade-off solution. By exploring all these optimization possibilities, we ensure the discovery of the complete Pareto front.

The implementation of the method for *FP* is apparent. There are only two objectives thus only two sequences of the function, first the battery optimization and the charging optimization second and vice versa. Then, one of the optimal values of an objective can be relaxed, i.e. by adding an extra charging station. The other objective is optimized once more to potentially uncover trade-off solutions. The charging stations are in a lower volume than the number of batteries, therefore this objective can be relaxed from its marginal optimal to the optimal reached by the optimization where the batteries are prioritized. Algorithm 4.7 shows the full method as applied to *FP*. This tactic uncovers the full $\mathcal{PF}^*$ in a few *FP* optimization runs.

Algorithm 4.7 (One-by-One optimization method for *FP* to construct the Pareto front)**.** *First, we start with battery priority, then charging station priority. Then, all the intermediate options are checked.*

1. *Battery priority optimization (BPO):*
    (a) *Optimize* $f_1(x) = \sum_b y_b$ *given infinite charging stations, giving an optimal solution* x^**, and set* $B_1 = f_1(x^*)$*.*
    (b) *Set* $y_b = 1$ *for* $b \in \{1, \cdots, B_1\}$ *and* $y_b = 0$ *for* $b > B_1$*. Optimize* $f_2(x) = M$*, giving an optimal solution* x^**, and set* $M_1 = f_2(x^*)$*.*
2. *Charging station priority optimization (CSPO):*
    (a) *Optimize* $f_1(x) = M$ *given a large enough number of batteries, giving an optimal solution* x^**, and set* $M_2 = f_1(x^*)$*.*
    (b) *Set* $M = M_2$*. Optimize* $f_2(x) = \sum_b y_b$*, and set* $B_2 = f_2(x^*)$*.*
3. *Intermediate solutions, for* $M_1 < M < M_2$ *(counted by* j*) do:*
    (a) *Set* $M_{j+2} = M$*.*
    (b) *Optimize* $f_{j+2}(x) = \sum_b y_b$*, giving an optimal solution* x^**, and set* $B_{j+2} = f_{j+2}(x^*)$*.*
4. *Set* $\mathcal{PF}^* = \{(M_l, B_l) | l \in \{1, \cdots, j+2\}\}$

4.2.2 Weighted Sum Method

Another often-used method to find Pareto optimal solution is the weighted sum method. It involves assigning weights to the different objective functions and combining them into a single objective function that the optimization can solve. The assignment of the weights is done on the preference of the decision maker, often the prices or penalties of certain objectives. We introduce weights $w_i \in [0.1]$ for all objective functions f_i in problem 4.1 and transform the objective function $\min F(x)$ into $\min \hat{F}(x) = \sum_{i=0}^{d} w_i f_i$.

In the context of *FP*, its objective function transform to be:

$$\min w_0 M + w_1 \sum_b \cdot y_b. \tag{22}$$

The solution of this optimization gives the Pareto optimal point according to the decision maker's preference. The Pareto front can be generated by systematically varying the weights. However, Marler and Arora [2004] state that if the feasibility criterion space Z is non-convex, there may be Pareto optimal solutions that the method cannot find. *FP* is a scheduling problem and therefore contains integer decision variables. This implies that the Z of *FP* is non-convex and therefore some Pareto optima might not be found.

Preferably, the weights should be set to their respectable investment costs. The costs would most likely fully influence the preference of the decision-maker. However, these costs are unknown and estimation is out of scope for this research. It is hard to estimate which weights push the optimization to what solution. The one-by-one solution is a method to estimate the effect of the weights on the system and is therefore more suitable for this problem. For this reason, we select the one-by-one method as our method of choice. The following chapters will only use the one-by-one optimization method to deal with the bi-objective optimization problem *FP*.

5 Methodology

The two optimization problems described in Chapter 2 can be represented as stated in Appendix B.3 and B.4. These representations are specifically made to describe a mixed integer linear problem (MIP). Given a timeline and all the parameters (without any uncertainty), a solver can directly solve the instance of both problems and provide an optimal answer with a schedule that satisfies the constraints of the problems.

The schedule found when solving the problem without uncertainties can be seen in a probabilistic sense as a schedule that reaches the objective in the most likely scenario. All the vessels have to use exactly the expected percentage of battery power and arrive precisely at the expected time. However, this is rarely the case in real life. The energy consumption is largely correlated with the mass of the cargo that is transported by the vessels, which may not be known at the time of scheduling. The expected arrival times of the ships are heavily dependent on vessel delays, such as closed bridges or traffic. Also, both the energy consumption and the arrival times are dependent on the speed the vessel is sailing, which is not influenced by the party that schedules the batteries.

Therefore, the question arises: which of the schedules that have the potential to reach the optimal objective is most optimal in a probabilistic sense? Optimal can be viewed in two ways:

1. What schedule produces the most expected profits? (Profitability)
2. What schedule is the most likely to succeed? (Robustness)

Gorissen et al. [2015] states that there are two main approaches to dealing with data uncertainty in optimization, these are *stochastic* and *robust* optimization. The key difference between the two approaches is that stochastic optimization assumes that the underlying distribution is known or can be estimated whereas robust optimization assumes solely that the uncertain data resides inside of a so-called uncertainty set. Stochastic programming is the subset of stochastic optimization where the problems can be modeled as a linear or a nonlinear program as is the case for our problem. Furthermore, integer programming often fails to converge for large instances of the problem. In literature, heuristics are often used to solve a problem or enhance a viable solution, these will be left out of this research.

This chapter is divided into three sections. There are six optimization methods proposed, which all try to find the optimal number of batteries, charging stations, and grid balancing stints in different contexts. The first section describes the general MIP formulation that is used to find the optimal of all objectives in a deterministic case. The second section explains three stochastic programming methods which are extensions to the MIP formulation. These methods are used for scheduling problems that deal with known uncertainties. Direct Monte Carlo simulation optimizes the problems of testing a high volume, rolling horizon optimizes the problems in a realistic time-shifting scenario, while the probabilistic constraint method assumes the problem solutions and finds the most robust schedule. In the third section, two robust techniques are described, both of which consider the most unfavorable values of uncertainty characteristics before optimizing the problems.

5.1 Mixed integer linear programming

An integer linear program (ILP) is an extended linear program where a decision variable vector x consists solely of integers. A method to solve an ILP is called 'branch and bound'. This method is based on a solving algorithm used for linear programming that was proposed by Dantzig in 1947, called the simplex method (Dantzig and Thapa [1997]). This algorithm makes a descent along the outside edges of the polyhedral set of a linear problem to reach the optimal solution. The decision variables represented by an integer range are handled using a technique known as branching. In this approach, the decision variables are structured in a decision tree, where each branch corresponds to a possible value the variable can take. By traversing the branches, the objective function for all the possibilities for the decision variables can be explored.

As the input size increases, the computational cost of visiting all the branches increases. Therefore, branch and bound uses LP relaxations, the continuous variant that can be solved by the simplex method, to generate an optimal bound for all unvisited branches. This represents the bound component of the branch and bound. All the unvisited branches are stored in a list ordered by the LP relaxation bound and are explored by visiting the branch with the highest potential. When a solution is found with an objective value that is better than the optional bound for a specific branch, the solutions in that branch will always be worse than the found solution. Thus, the branch is cut from our list and no longer explored. Given enough time all the potential branches will be cut and the optimal solution will be found. This is a direct algorithm to solve the problem, therefore by the complexity of the problem, it cannot converge in polynomial time over the input size. The decision tree grows exponentially with the input size x in branch and bound.

A mixed integer linear program (MIP) allows a combination of inputs that are integers and continuous. The branch and bound algorithm still suffices to solve a MIP, the continuous inputs do not show up in the decision tree and they do not have to be LP relaxed, and are optimized by the simplex method used in branch and bound.

Another popular method for solving ILPs and MIPs is the cutting-plane algorithm. The basic idea of cutting planes as described by Pióro and Medhi [2004] is to iteratively build upon the LP relaxation by adding additional constraints to eliminate the fractional solutions. These additional constraints are called 'Parameter Cuts'. The cuts are often derived from a theoretical analysis of the problem. In each iteration, the LP relaxation is solved and the violated cutting planes are identified and added to the formulation. This ensures that the fractional solutions that the simplex method finds are cut off until eventually an integer solution is found.

Branch and cut is a framework where branch and bound and cutting planes are combined. This framework iteratively uses both a branch and bound and a cutting planes step. The initial problem is branched into a selected decision variable and both branches are bounded by their LP relaxation and added to the list to be further explored. Then, a cutting plane is added to the system to cut off fractional solutions. As in branch and bound, if an intermediate solution is found better than the upper bound the respective branches are cut from the list. These steps are repeated until an optimal integer solution is found or the problem is proven to be infeasible.

The MIP solving method used in this research starts with a plane cut initialization and then solves the resulting MIP with branch and bound.

5.1.1 Parameter Cuts

The model described in Chapter 2 tends to have a large difference in the LP bounds and the intermediate/optimal solutions. This difference often leads to slow convergence using a branch and bound method. The speed of the method can be increased by implementing parameter cuts that make the LP relaxation bound of a branch tighter to its potential optimal solution objective as done in the cutting planes framework. The cuts consist of extra variables and/or constraints that increase the cut ratio of the branches. The following parameter cuts strengthen the LP bounds for this specific problem. There is a large deficit between the *FP* objectives functions 13, 14 and their LP counterparts. The following continuous variables and constraints push the LP relaxations of B and M closer to their integer solutions:

Extra Variables	Ranges	Descriptions
U_b^t	$[0, 1]$	A status variable that denotes if battery b is currently utilized by a vessel
I_b^t	$[0, 1]$	A status variable that denotes if battery b is currently idle

Table 5: Parameter Cut Variables

The following constraints are formulated with these variables:

$$U_b^t = \sum_{\hat{t}} x_b^{\hat{t}} \cdot A(\hat{t}, t), \quad \forall b, t \tag{23}$$

$$U_b^t + I_b^t \leq y_b, \quad \forall b, t \tag{24}$$

$$w_b^t \leq I_b^t, \quad \forall b, t \tag{25}$$

The first constraint (Equation 23) sets the value of the sailing status of battery b at time t. The second constraint (Equation 24) pushes the LP relaxation of $\mathbf{y}$ to be higher and closer to its optimal solution. The last constraint (Equation 25) pushes the LP relaxation of M closer to its optimal solution. The full MIP formulations of both problems are found in Appendix B.5 and B.6.

5.1.2 MIP Pareto Optimization

As seen in Chapter 4, the Pareto front is constructed using the one-by-one optimization method. The exact strategy is described in Algorithm 4.7. The battery and charging station optimizations are respectively solved by the MIPs B.6 By applying this algorithm, a set $\mathcal{PF}^*$ of solutions $(M, B)^*$ is found. One of the solutions in the $\mathcal{PF}^*$ is used as input for the optimization of RP.

5.2 Stochastic Programming

Stochastic programming is an extension of the mathematical programming framework that addresses uncertainty. The goal of stochastic programming is to find the optimal solution to the problem considering the randomness of certain parameters. Stochastic optimization offers the advantage of finding solutions that are more flexible and robust, enabling better decision-making in realistic and uncertain situations for operators.

There are typically two primary sources of uncertainty that are considered throughout the optimization: discrete scenarios and continuous random variables. Direct Monte Carlo simulations and Rolling Horizon use Monte Carlo sampling, which creates deterministic scenarios generated under the uncertainties included in the scope. These scenarios can be solved by the MIPs discussed in the previous section. The probabilistic constraint approach takes the full uncertainty of a random variable into account throughout the optimization. This leads to an extension of the MIP, where an additional constraint is introduced to limit the probability of failure within the specified uncertainty.

5.2.1 Direct Monte Carlo Simulation

Direct Monte Carlo Simulation (*DMCS*) makes use of the law of large numbers formally stated in Appendix A in Theorem A.17. By random sampling from the uncertainty distribution (Monte Carlo simulating) a sampling size N of scenarios is generated. These scenarios are deterministic and can be solved by the methods discussed in the previous section. By evaluating a large scale of scenarios, inference can be drawn on the expected results and the robustness of the decisions made by the model.

The sampling size N is very crucial to be able to verify the results. If the N is too small, this may lead to unreliable results since a large part of the input space may be underrepresented. However, if the value of N is too large, it can significantly increase the computational costs of the method. Therefore, multiple sample sizes are tested, $N = 10, 100, 1000$.

The size of the event time-space is adjusted throughout the different sample sizes to keep the optimization method within a reasonable time span. Also, if the number of uncertain parameters is increased, the sampling region grows exponentially. Then, the sample size needs to be exponentially higher to stay appropriate.

A drawback from direct *DMCS* is the substantial computational costs involved. The approach requires solving the deterministic case multiple times, which, in our scenario, involves solving a MIP problem N times. Furthermore, as the input size increases, the solving time of the MIPs increases exponentially. Consequently, *DMCS* solving times grow even faster if the input size is increased. For these reasons, the scalability of *DMCS* is very limited.

5.2.2 Rolling Horizon

The Rolling Horizon (*RH*) method is also based on Monte Carlo Sampling. However, it addresses one of the main limitations of Direct Monte Carlo simulations, which is the scalability of the problem. This limitation is due to the need to perform a substantial number of simulations to obtain reliable results. This can be computationally intensive and limits the scale of the problem due to time constraints.

In contrast, the *RH* method offers an extension that has the potential to provide insights over

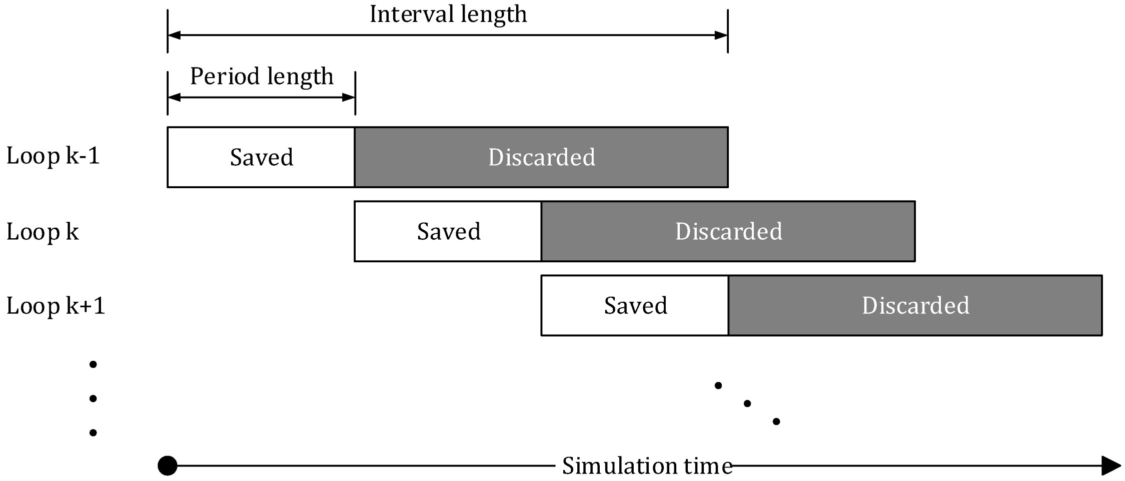


Figure 6: Schematic diagram of rolling horizon (Erichsen et al. [2019])

longer periods. Instead of conducting a massive number of simulations upfront, the *RH* approach involves dividing the problem into shorter time horizons. Each horizon is optimized independently, taking into account the current state of the system and available information. As time progresses, the horizon is rolled forward, incorporating new data and fixing the decisions made that do not transfer to the next scenario. Figure 6 depicts a diagram of the creation methods of the different horizons. By iteratively optimizing over shorter horizons and updating the solution over time, the *RH* method allows for a more scalable approach for handling complex problems.

Furthermore, the rolling horizon method closely mirrors the decision-making process in real-world scenarios. Operators have to make contractual decisions in a timely fashion and not all at the start of the week. Fixing decisions in different horizons simulates this process. Moreover, as the timing of an event gets closer, the uncertainties associated with this event become more predictable. This can also be incorporated in *RH* by adjusting the probability distributions of the uncertainties when the associate event draws closer.

The horizons are all optimized separately and combined to form a single solution for the full timeline. By stinging multiple stand-alone optimizations together, the guarantee of an optimal outcome for the whole time-space is lost. The system loses oversight of system-wide problems or opportunities. Furthermore, due to a large number of optimizations conducted, the method is prone to increasing convergence times. Moreover, infeasabilies may occur due to non-optimal intermediate solutions, which create a different starting situation for the following horizon, which may become infeasible.

All in all, *RH* enables decision-makers to make adaptable and dynamic choices while still considering the uncertain and evolving nature of the problem. However, it's important to note that certain cases might require extended computational time and could result in suboptimal or potentially infeasible outcomes.

5.2.3 Probabilistic Constraint

Probabilistic Constraints (*PC*) extend the MIP formulation to only allow solutions with a degree of robustness. The feasibility space of the problem is cut short by not allowing non-robust solutions.

The main advantage is that the solution found by *PC* guarantees a solution that is robust up to the defined limit and optimal in the smaller feasibility space.

However, similar to the previous two methods, PC contributes to additional computational costs in the MIP optimization. Another disadvantage of using PC is that the probabilities that are needed to constrain the robustness of an optimization are usually hard to compute. This method is only used on for the uncertainty in energy consumption E_t, because the probabilities of bounding the robustness are relatively easy to calculate. The counterpart for applying this method to the arrival time uncertainty is considered outside of the scope of this research.

In the implementation of our problem, we assume that the number of container batteries B, charging stations M and revenue R are found by the deterministic MIP formulation. Then, the goal of probabilistic constraint optimization is to find the optimal schedule in the sense of robustness.

Assume that $E_t \sim N(\mu_t, \sigma_t)$ distributed. The probabilistic constraint used is limiting the chance of failure at any time t by a factor α. The new objective is minimizing this factor α, to minimize the probability of failure at any time. The results in the following minimization problem:

$$\min_{\alpha} \mathbb{P}(\text{Failure at time } t) < \alpha, \forall t. \tag{26}$$

Failure occurs if there is no battery with enough charge to facilitate the vessel with energy, i.e.

$$\min_{\alpha} \mathbb{P}(c_b^t < E_t) < \alpha, \forall b, t. \tag{27}$$

α must be greater then all b, t and is therefore equivalent to being greater than the largest, i.e.

$$\min_{\alpha} \max_{t,b} \mathbb{P}(c_b^t < E_t) < \alpha. \tag{28}$$

Assume that route of battery b_1 incorporates the task that starts on time t. Define the route of b_1 to be $R_{b_1} = \{t_1, \cdot, t_j, t, \cdots, t_m\}$. Define $OPL_{b_1}^r$ to be the charge added to battery b_1 in-between the tasks of route R_{b_1}. Assume that the starting charging level equals 1. Then, $c_{b_1}^t$ can be calculated to be:

$$\begin{aligned} c_{b_1}^t &= c_{b_1}^0 - E_{t_1} + OPL_{b_1}^1 - E_{t_2} + OPL_{b_1}^2 - \cdots - E_{t_j} + OPL_{b_1}^j \\ &= 1 - E_{t_1} + \sum_{t_1 < t_a < t_2} d \cdot \Delta_{t_a} \cdot w_{b_1}^a - E_{t_2} + \sum_{t_2 < t_a < t_3} d \cdot \Delta_{t_a} \cdot w_{b_1}^{t_a} - \cdots - E_{t_j} + \sum_{t_j < t_a < t} d \cdot \Delta_{t_a} \cdot w_{b_1}^{t_a} \\ &= 1 + \sum_{r \in R_{b_1} | r < t} (-E_r + \sum_{r < t_a < r+1} d \cdot \Delta_{t_a} \cdot w_{b_1}^{t_a}). \end{aligned} \tag{29}$$

The failure occurs when $E_t > c_b^t$ at time event t, therefore

$$\begin{aligned} E_t &> 1 + \sum_{r \in R_{b_1} | r < t} (-E_r + \sum_{r < t_a < r+1} d \cdot \Delta_{t_a} \cdot w_{b_1}^{t_a}) \\ \sum_{r \in R_{b_1} | r \leq t} E_r &> 1 + \sum_{r \in R_{b_1} | r < t} \sum_{r < t_a < r+1} d \cdot \Delta_{t_a} \cdot w_{b_1}^{t_a}. \end{aligned} \tag{30}$$

Now using that $E_t \sim N(\mu_t, \sigma_t)$, we know that the sum of a normal distributed function is also normally distributed. Therefore,

$$\sum_{r \in R_{b_1} | r \leq t} E_r \sim N(\sum_{r \in R_{b_1} | r \leq t} \mu_r, \sum_{r \in R_{b_1} | r \leq t} \sigma_r).$$

This distribution can be scaled and shifted to a parameter z that is standard normally distributed:

$$z = \frac{\sum_{r \in R_{b_1} | r \leq t} E_r - \sum_{r \in R_{b_1} | r \leq t} \mu_r}{\sum_{r \in R_{b_1} | r \leq t} \sigma_r} \sim N(0, 1). \tag{31}$$

We scale the right part of Equation 30 with the same values to create a critical value $\beta^t_{b_1}$ where failure occurs if $z > \beta^t_{b_1}$. This gives the following critical value for battery b_1 at time-event t:

$$\beta^t_{b_1} = \frac{1 + \sum_{r \in R | r < t} \sum_{r < t_a < r+1} d \cdot \Delta_{t_a} \cdot w^{t_a}_{b_1} - \sum_{r \in R_{b_1} | r \leq t} \mu_r}{\sum_{r \in R_{b_1} | r \leq t} \sigma_r}. \tag{32}$$

This can be generalized to all batteries b, where $\beta^t_b = 0$ if battery b is not allocated at time-event t and equal to Equation 32 if the battery is allocated. Also, we use the allocation $x^r_b = 1$ if and only if $r \in R_b$. This gives the following formulation:

$$\beta^t_b = \frac{1 + \sum_{t_1 < t} d \Delta^{t_1+1} w^{t_1+1}_b - \sum_{t_a | t_a < t} \mu_a x^{t_a}_b - \sum_{t_g | t_g < t} \mu_g x^{t_g}_b}{\sum_{t_a | t_a < t} \sigma_a x^{t_a}_b + \sum_{t_g | t_g < t} \sigma_g x^{t_g}_b}. \tag{33}$$

This formulation is nonlinear. To linearize this formulation, the average number vessel ($\bar{X}^t_a$) and grid balancing allocations ($\bar{X}^t_g$) before time event t are substituted in the denominator. These averages can be estimated by dividing the sum of all tasks done before time t over all the batteries in the system equally. This averaging does affect the outcome slightly but gives a valid linearization.

$$\beta^t_b = \frac{1 + \sum_{t_1 < t} d \Delta^{t_1+1} w^{t_1+1}_b - \sum_{t_a | t_a < t} \mu_a x^{t_a}_b - \sum_{t_g | t_g < t} \mu_g x^{t_g}_b}{\bar{X}^t_a \sum_{t_a | t_a < t} \sigma_a + \bar{X}^t_g \sum_{t_g | t_g < t} \sigma_g}. \tag{34}$$

Now we can limit all the β^t_b by introducing a lower bound β:

$$\beta \leq \beta^t_b, \forall b, t. \tag{35}$$

Then, Equation 28 can be rewritten as the following:

$$\min_{\alpha} \mathbb{P}(z > \beta) < \alpha. \tag{36}$$

This minimization is equivalent to maximizing β. This gives the following objective

$$\max \beta. \tag{37}$$

This objective can be incorporated into the MIP formulation of the RP problem. The new MIP formulation for RP can be found in Appendix B.8.

5.3 Robust techniques in stochastic programming

Formally robust optimization is set up using the following definitions of the uncertainty set and a general robust optimization (RO) problem.

Definition 5.1 (Uncertainty set)**.** The *uncertainty set* is the set of possible values of the uncertain parameters that are considered in the robust optimization problem. This set will be called $\mathcal{U}$, which could be split into the single dimension spaces $\mathcal{U}_i$ for each uncertainty parameter u_i for $i \in \{1, \cdots, m\}$.

Bertsimas et al. [2010] give the following formulation for the generalized robust optimization problem.

Definition 5.2 (General formulation of Robust optimizations problems)**.** Given an objective function $f_0(\mathbf{x})$ to optimize, subject to constraints $f_i(\mathbf{x}, \mathbf{u}_i) \leq 0$ with uncertain parameters, $\{\mathbf{u}_i\}$, the general Robust Optimization formulation is

$$\begin{aligned} &\text{minimize:} \\ &\qquad f_0(\mathbf{x}) \\ &\text{subject to:} \\ &\qquad f_i(\mathbf{x}, \mathbf{u}_i) \leq 0, \quad \forall \mathbf{u}_i \in \mathcal{U}_i, i = 1, \cdots, m \end{aligned} \tag{38}$$

Here $\mathbf{x} \in \mathbb{R}^n$ is a vector of decisions variables, $f_i : \mathbb{R}^n \to \mathbb{R} \forall I = 0, \cdots, m$ are functions, and the uncertainty parameters $\mathbf{u}_i$ take arbitrary values in the set $\mathcal{U}_i$.

The uncertainty sets of the energy consumption uncertainty and time arrival uncertainty are unknown. In this research, we formulate the uncertainty sets with assumptions. In reality, the uncertainty sets can be estimated by using confidence bounds on the empirical distribution that can be created from historical data. Two robust optimization-based approaches are formulated for the energy consumption- and time uncertainties, deliberately choosing the conditions that are unfavorably compared to most scenarios. The solutions of the methods described in this section are conservative and serve as a strict lower bound for the solutions that result from the stochastic programming methods.

5.3.1 Extreme Value Analysis Energy Consumption Uncertainty

Two dominating cases regarding energy consumption can be formulated by choosing the upper limit and the lower limit of the uncertainty sets. The upper limit of the uncertainty set of the energy consumption serves as a worst case. This is due to that the battery could always dump electricity within two-time events, therefore having too much energy in the batteries can never be a restriction on the objective values obtained from the optimization. However, when there is less charge in the batteries, they need time and a charging station docking to be recharged before being redeployed. In the context of the problem, it is beneficial for the batteries to contain as much charge as possible. Therefore, the lower limit serves as a best case scenario.

5.3.2 Value At Risk Analysis Arrival Time Uncertainty

The value at risk analysis for arrival time uncertainty considers that the extreme situations of the time arrivals occur simultaneously. We define a level of significance α, which can be used to construct a $100 \cdot (1 - \alpha)\%$ confidence interval with a lower and upper bound.

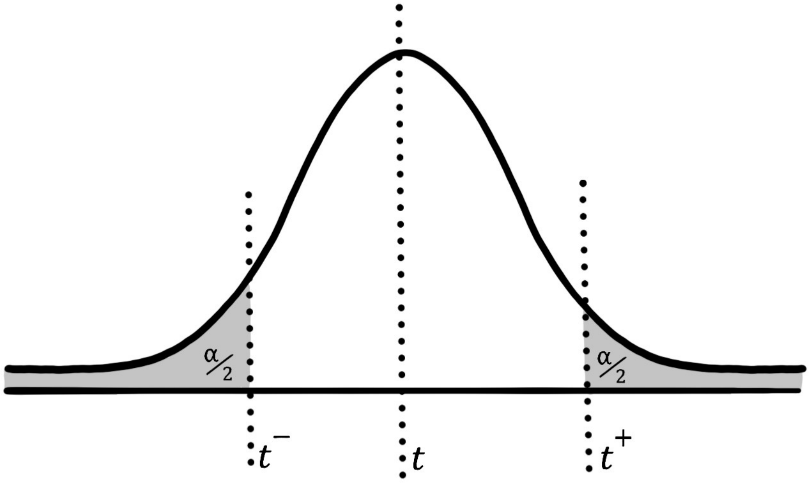


Figure 7: The arrival times are split according to the $100 \cdot (1-\alpha)$%-confidence interval (CI) of the distribution of the arrival time. This is showcased in the normal distribution, but it can be applied to all distributions that the arrival time may have, including an empirical distribution.

In this approach, the event space is divided into two sets, one consisting of all the lower bounds and the other of all the upper bounds. It assumes that, for all arrival times, a battery is needed at the earliest possible moment of arrival, which corresponds to events in the first set t^-. Conversely, the battery is returned at the last possible moment that the ship can arrive, which corresponds to events in the second set t^+. The interval between t_a^- and t_a^+ is revered to as *the critical interval* of arrival a. Figure 7 shows how the two new events are created for every arrival time. Note that the starting times of the grid balancing stints are known and therefore are not split and remain a single-time event.

Battery allocation decisions are only made on events that are part of the first set. All parameters related to the allocation of the batteries of the time events are only defined on the first set (t^-). This results in the following changes to their counterpart in Section 2.3:

Parameters	**Descriptions**
$\hat{A}_{t_1^-,t_2^+}$	The matrix that shows the relation of battery availability and time events in the value at risk scenario of the arrival time uncertainty.

Table 6: Changed parameters in value at risk formulation

Also, Constraint 1 is only defined for the time events of the form t^-:

$$\sum_b x_b^{t_a^-} = 1, \quad \forall t_a^- \tag{39}$$

This method gives a very conservative solution to both *FP* and *RP*. The smaller the chosen α the more conservative the solution becomes.

6 Experimental results

In this section, three case studies of the model will be evaluated[4]. The cases are optimized using the suitable methods described in Chapter 5 as well as the approximation algorithm described in Section 3.2.

1. Deterministic Case: A single port deterministic system with simulated arrival times.
2. Energy Uncertain Case: A single port system with uncertainty in energy consumption by both vessels and grid balancing stints.
3. Time Uncertain Case: A single port system with uncertainty in arrival times of the cargo vessels.

The complexity of the model is increased in the case studies by adding a layer of uncertainty. The first case includes a sensitivity analysis on a base case to explore the implications for the objectives and the computational time of the deterministic model. The first case forms the baseline for the following cases where the Base Case defined also serves as the starting point for the following cases.

Every case has the same basic structure:

- Time horizon of a single week,
- Five electrified vessels that visit the port to swap their batteries with a return time of approximately six hours,
- Forty-one grid balancing stints that last four hours each and are non-overlapping.

[4] The processor used to run all experiments is 11th Gen Intel(R) Core(TM) i7-1185G7 with 16,0 GB RAM

6.1 Case 1: Deterministic Case

For the first case, we consider a fully deterministic system. The case consists of a single charging port system with simulated arrival times and fixed energy consumption. The system contains 5 ships that all have an estimated voyage duration of 6 hours. Firstly, a base case is formulated. Secondly, the optimization strategies used in this case are listed. Thirdly, the results of the base case are shown. Fourthly, for all input parameters, a sensitivity analysis is performed. Then, the consequences of multiple and overlapping trade opportunities are discussed. Finally, computational analysis is done on the performance of the optimizations when the time horizon or the fleet size is increased.

6.1.1 Base Case

A use-case scenario needs values for all the parameters described in Section 2.3. The following base-case scenario is defined.

Definition 6.1 (Base Case (BC))**.** In this scenario, the arrival times of the five vessels operating in the system are simulated individually. The timeline chosen for this scenario spans from 1 January 2019, 00:00, to 8 January 2019, 23:59. The following parameter choices are made:

Parameters	Value Base Case
t_{min}	1 January 2019, 00:00
t_{max}	8 January 2019, 23:59
C_t^-	0.8 for t_a, 0.3 for t_g
C_t^+	1.0 for t_a, 0.7 for t_g
$[C_t^-, C_t^+]$	[0.8,1.0] for t_a, [0.3,0.7] for t_g
T_g	Every 4 hours
d	0.0001 (battery is charged in around 2 hours and 47 min)
e	10^-6 (battery is self-discharged in around 116 days)
S_b	The first 5 batteries start respectably on the vessels
c_b^0	1 for all battery b
R_{t_g}	1 for all t_g
E_t	0.7 for all t_a, 0.1 for t_g

Table 7: Parameters Base Case.

The return time of each vessel follows a normal distribution in this scenario with an expectation of 6 hours and a variance of 1 hour, the simulations that end up outside of the one-week horizon are omitted from the case. The simulated arrival times can be seen in Table 26 in Appendix D and are plotted in Figure 8. Figure 9 shows the same timeline with the grid balancing events added. The parameters Δ_t and A_{t_1,t_2} can be derived from the timeline, for the extracted Δ_t see Table 28 in Appendix D.

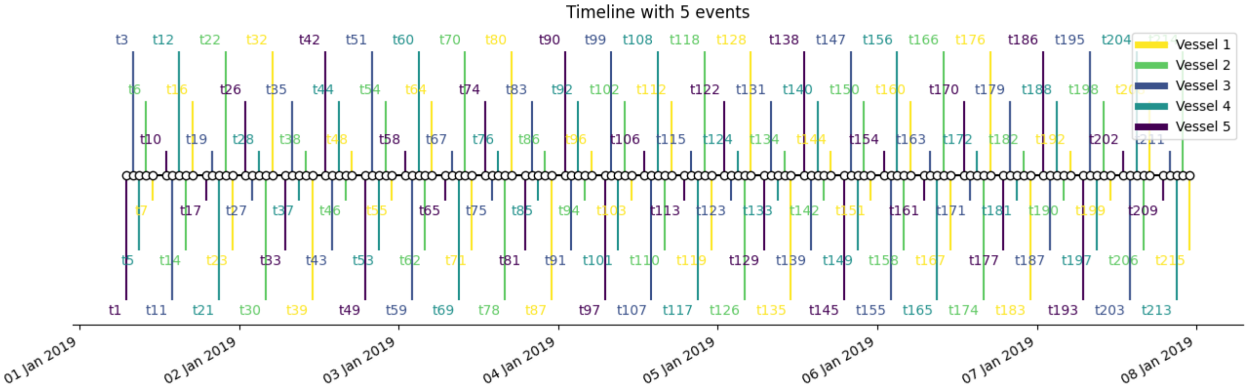


Figure 8: The simulated event time-space for the deterministic Base Case without grid balancing with only the arrival times of the vessels. The timeline is created by individually simulating the return times of all vessels. The return times are all distributed normally with an expectation of 6 hours and a variance of an hour. The full timetable can be found in Appendix D in Table 26.

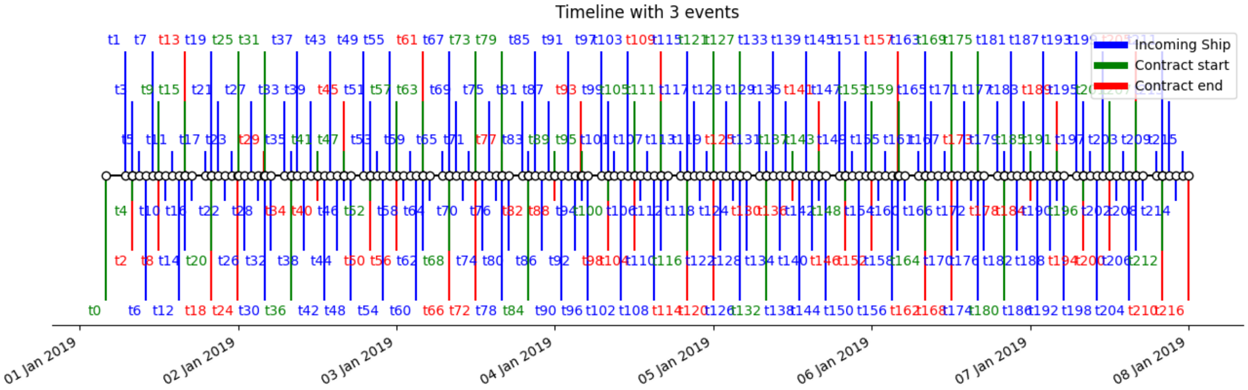


Figure 9: The full simulated event time-space for the deterministic Base Case with grid balancing. This figure combines Figure 8 with non-overlapping grid balancing stints, every 4 hours. Note that the timeline consists of 137 ship arrivals and 41 grid balancing stints. The timing of the grid balancing stints can be found in Appendix D in Table 27.

6.1.2 Optimization Strategies

As described in Chapter 2, multiple objectives are taken into consideration. All the optimizations except for the approximation[5] are solved with mixed-integer linear programming[6] (MIPs). The following methods are used for the optimization of the first case:

1. Approximation Algorithm (AA): Polynomial-time algorithm based on the first-fit algorithm (Algorithm 3.13). This algorithm estimates the number of batteries and charging stations. All tasks, arrival times and grid balancing stints, that are included in the input are treated as compulsory. By fixing the charging stations the method can also be used to estimate the Pareto Front.

2. Battery Optimization (BO) : Objective function (13) subject to constraints (1 - 12) where the variable M is set as a parameter (B.5 with first objective).

3. Charging Station Optimization (CSO): Objective function (14) subject to constraints (1 - 12) where the variables y_b is set as a parameter (B.5 with second objective).

4. Revenue Problem Optimization (RPO): Objective function (15) subject to constraints (1 - 12) where both the M and y_b are set as parameters (B.6).

The first optimization method is a polynomial-time approximation method, based on the first-fit algorithm for bin packing. The second and third optimizations are solved using the timeline without grid balancing because there is no benefit in using any grid balancing. These two objectives together are combined in the *one-by-one* method to create the *Pareto front*, as explained in Chapter 4. The extremities of the *one-by-one* method are Battery Priority Optimization (BPO), first BO and then CSO, and Charging Station Priority Optimization, first CSO and then BO.

The last objective considers the optima of the first optimization objectives and optimizes for the most revenue.

6.1.3 Results Base Case

This section shows the results of the Base Case described in Section 6.1.1 using the methods listed in the previous section, and is divided into three parts. First, the results of the Approximation Algorithm are shown, this includes the number of expected batteries and charging stations and an estimation of the Pareto Front. Second, the Pareto front ($\mathcal{PF}^*$) of the one-by-one method is presented. Finally, the results of the Revenue problem are discussed for a Pareto optimal point used for the batteries and charging stations, while also considering an extra unit for one or both of these infrastructures.

The approximation algorithm approximates the number of batteries and charging stations necessary to be used in the Base Case. The approximation algorithm finds that the number of batteries B is equal to 9 and the number of charging stations M is equal to 3.

The schedule in Figure 10 validates the feasibility of a solution with this specific number of batteries and charging stations. Additionally, Figure 11 displays the usage of the charging station within the schedule, while Figure 12 illustrates the appropriate charge levels of the 9 batteries throughout the timeline in this schedule. This schedule appears to be an excellent fit for the Base Case without grid balancing.

[5] The approximation algorithm is run in Python 3.9.

[6] The optimizations are done in AIMMS software using the CPLEX 20.1 solver.

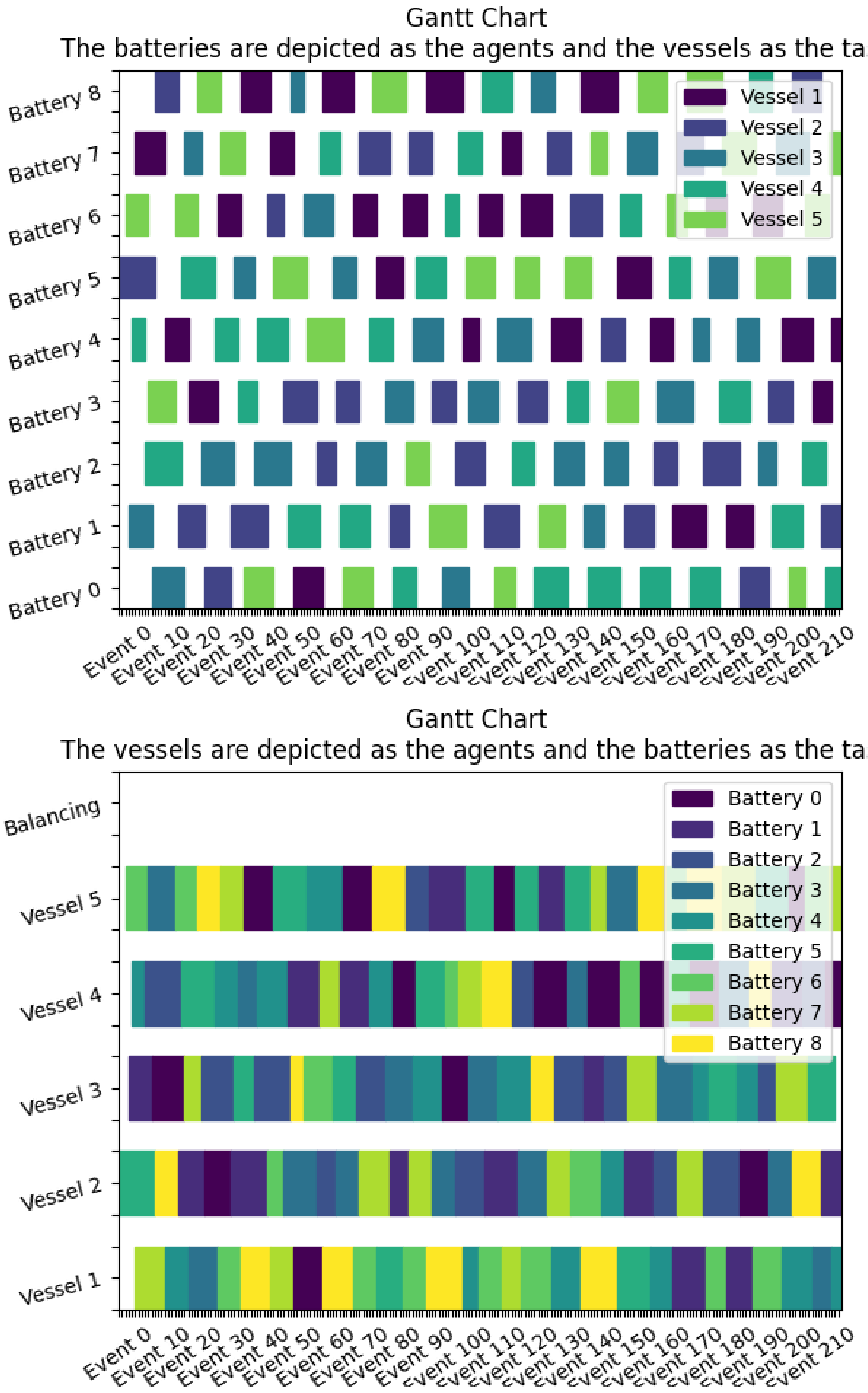


Figure 10: Gantt charts of the solution of the Base Case using the Approximation Algorithm 3.13. Note that the time axis is event-based rather than time-based. The timeline in Figure 9 presents the event time relation.

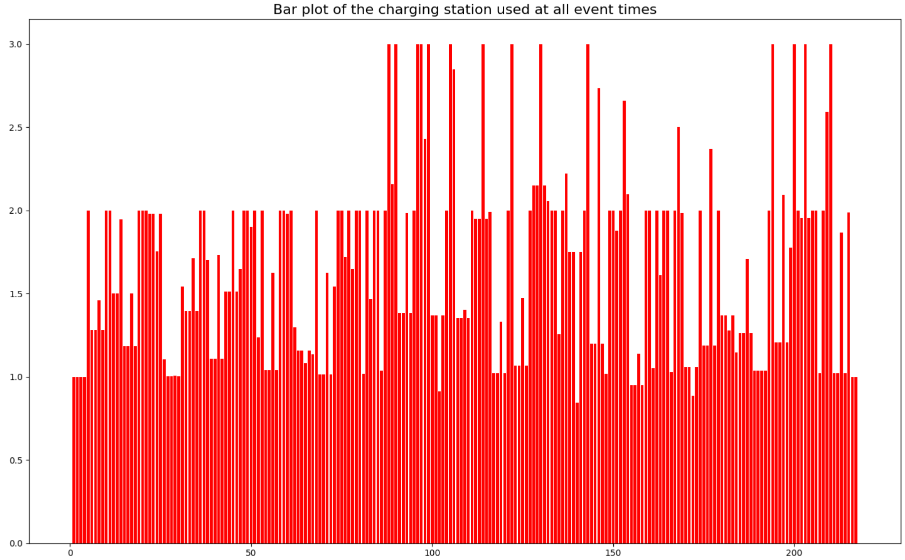


Figure 11: The utilization of charging stations between the time events for the Base Case are computed using the approximation algorithm.

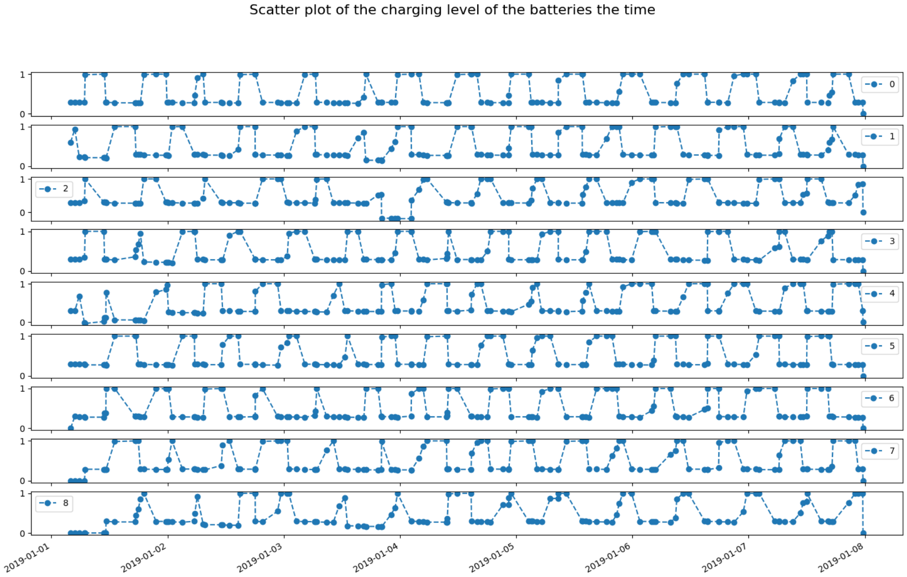


Figure 12: The charging levels of all the batteries for the Base Case are determined using the approximation algorithm.

Table 8 shows the estimation of the Pareto front using an approximation algorithm with a fixed number of charging stations. The one-by-one approach for the Feasibility Problem (FP) finds the values shown in Table 7 as the optimal values for FP. Figure 13 shows an example schedule found by BPO where the optimal values for FP are 9 batteries and 2 charging stations with Figure 14 and 15 the matching charging station usage and charging levels. The following *Pareto front* is found for FP. The Pareto front of the problem is likely to be:

$$\mathcal{PF}^* = \{(\mathbf{137}, \mathbf{0}), (\mathbf{44}, \mathbf{1}), (\mathbf{9}, \mathbf{2})\}. \tag{40}$$

The values accommodating this result are found in Table 9, the entry with 137 batteries can be logically deduced, there are no batteries that can be shared throughout the tasks, therefor a battery is needed for all the 137 arrival times in the system.

Notice the solution of the BPO depicted in Figure 13 and the AA solution shown in Figure 10 differ in the spaces that open up throughout the schedule. These spaces might be exploited by the operators to be used for grid balancing.

Table 8: Estimate $\hat{\mathcal{PF}}$ of Pareto front ($\mathcal{PF}^*$) calculated by the Approximation Algorithm 3.13 with a fixed number of charging stations.

$\hat{\mathcal{PF}}$		Computation
B	M	Time (s)
9	3	1.371
14	2	2.023
81	1	12.400
137	0	22.865

Table 9: The Pareto front ($\mathcal{PF}^*$) by the one-by-one method 4.7 for FP MIPs.

$\mathcal{PF}^*$		Computation	Gap	Note
B	M	Time (s)		
9	2	644.313	0 %	Optimal
44	1*	3600.00	4.5%	Best solution
137	0	-	-	Too large

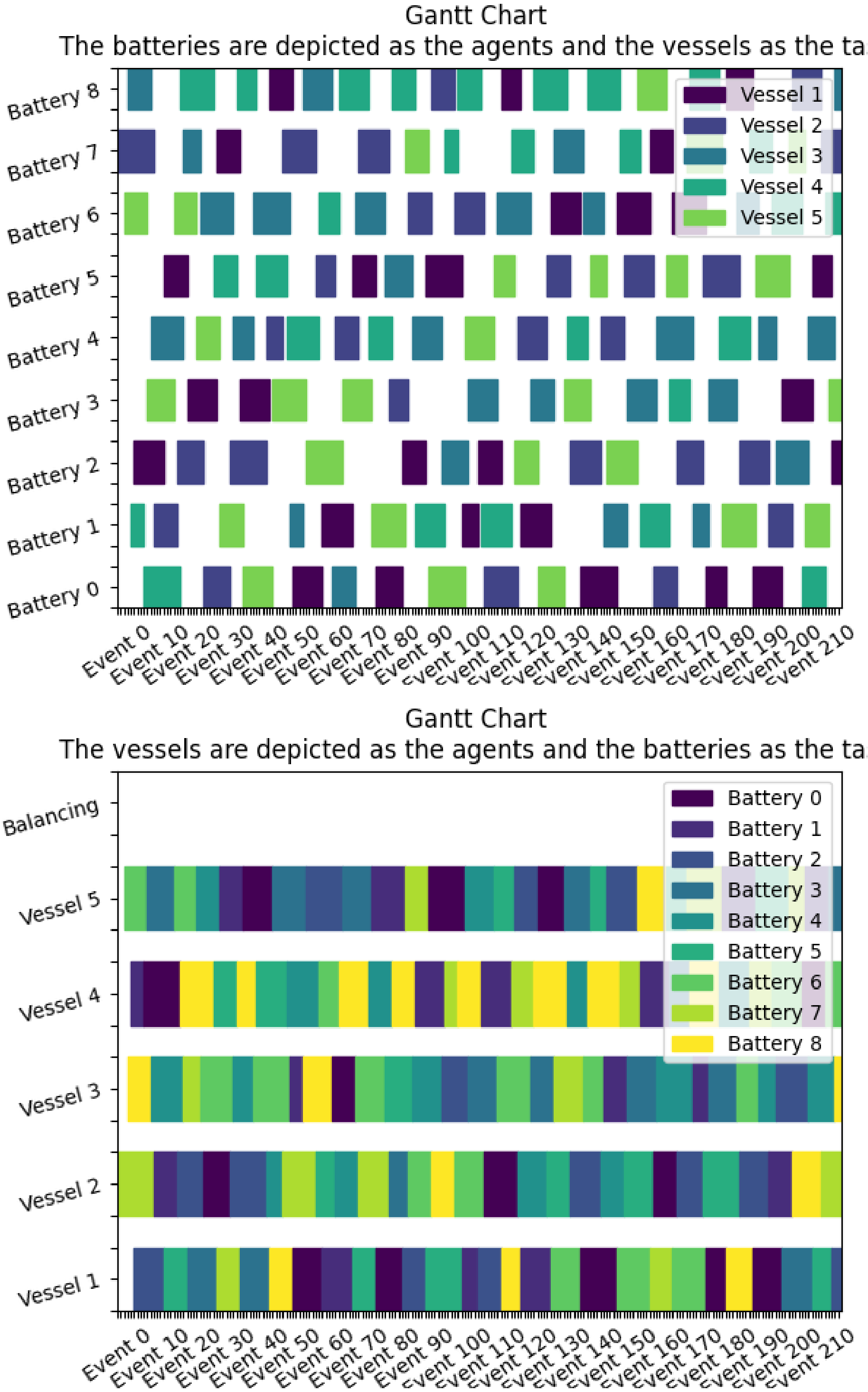


Figure 13: Gantt Charts Depicting outcomes for the Base Case using Battery Priority Optimization for the Feasibility Problem (FP) with parameter cuts (B.5). Note that this figure is event-based rather than time-based. The event time relation is depicted in the timeline shown in Figure 9.

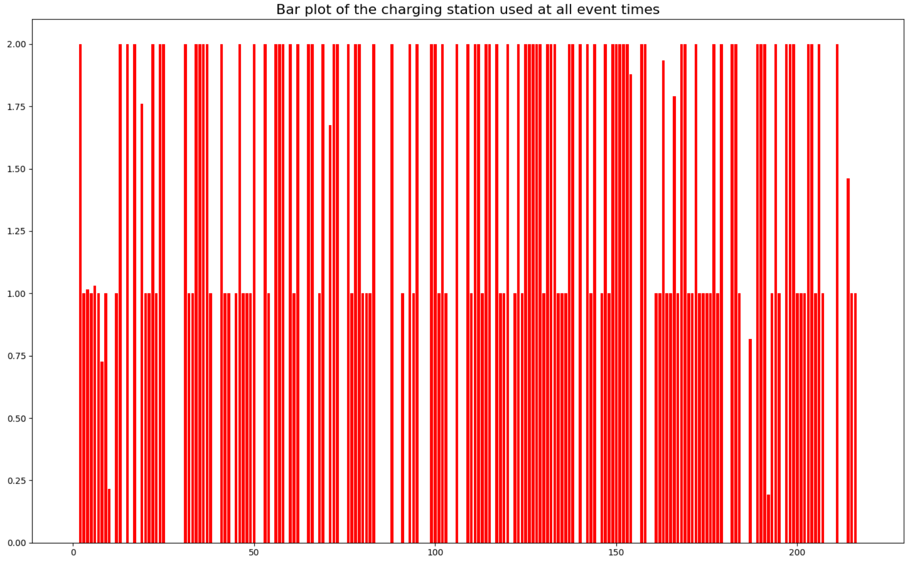


Figure 14: The charging levels of all the batteries for the Base Case are determined using the Battery Priority Optimization.

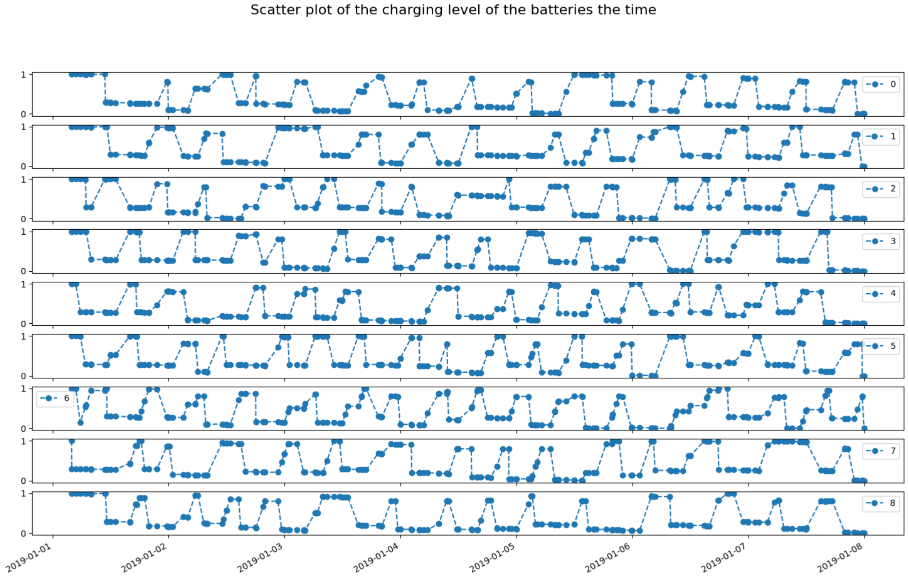


Figure 15: The utilization of charging stations between the time events for the Base Case are computed using the Battery Priority Optimization.

The approximation algorithm can also be used with grid balancing stint included in the input. Then, the grid balancing stints are treated as compulsory and the algorithm estimates 10 batteries and 4 charging stations. Figure 16 shows the results. This figure highlights the vulnerabilities in this approach, evidenced by the presence of seemingly redundant batteries and charging stations and extended periods of battery inactivity. This shows the necessity of a more complex method for choosing grid balancing stints.

To analyze *RP* we first assume that $(B, M) = (9, 2)$, then RPO finds the solution where a maximum of 20 grid balancing stints are used. Figures 17-20 show all facets of an example schedule where 20 grid balancing stints are carried out.

Given some degree of freedom in the batteries or charging stations, the optimization selects more grid balancing stints. Table 10 shows the results for a single degree of freedom for both the number of batteries and the number of charging stations. The scenario with an extra battery and an extra charging station can accept all balancing stints. The schedule that resulted from *FP*, depicted in Figure 13, can be implemented. Then, the additional battery and charging station remain idle and can be solely committed to grid balancing. This result is captured in Table 10 where the (B_{+1}, M_{+1}) accepts all grid balancing stints. The table also shows that the constraining infrastructure to accept more grid balancing stints is the number of charging stations. This indicates to the port operator that adding an extra charging station in some scenarios is economically viable.

Figures 17 and 18 show that there are still some open spaces in the battery schedules. This gives the impression that even more grid balancing might be possible. These white spaces occur while there is already a battery committed to balancing. Therefore, we are likely to see even more grid balancing if two batteries may be committed to grid balancing simultaneously. Section 6.1.5 explores these options.

Table 10: Results *RP* of the Base Case with different number of batteries (B) and charging stations (M). The starting values of the B and M are based on the The +1 denotes an extra battery or charging station on top of the previously listed optima.

Infrastructure	Amount	Revenue	t_{RPO} (s)
(B, M)	(9,2)	20	2357.047
(B_{+1}, M)	(10, 2)	21	2068.75
(B, M_{+1})	(9, 3)	38	248.797
(B_{+1}, M_{+1})	(10, 3)	41	14.594

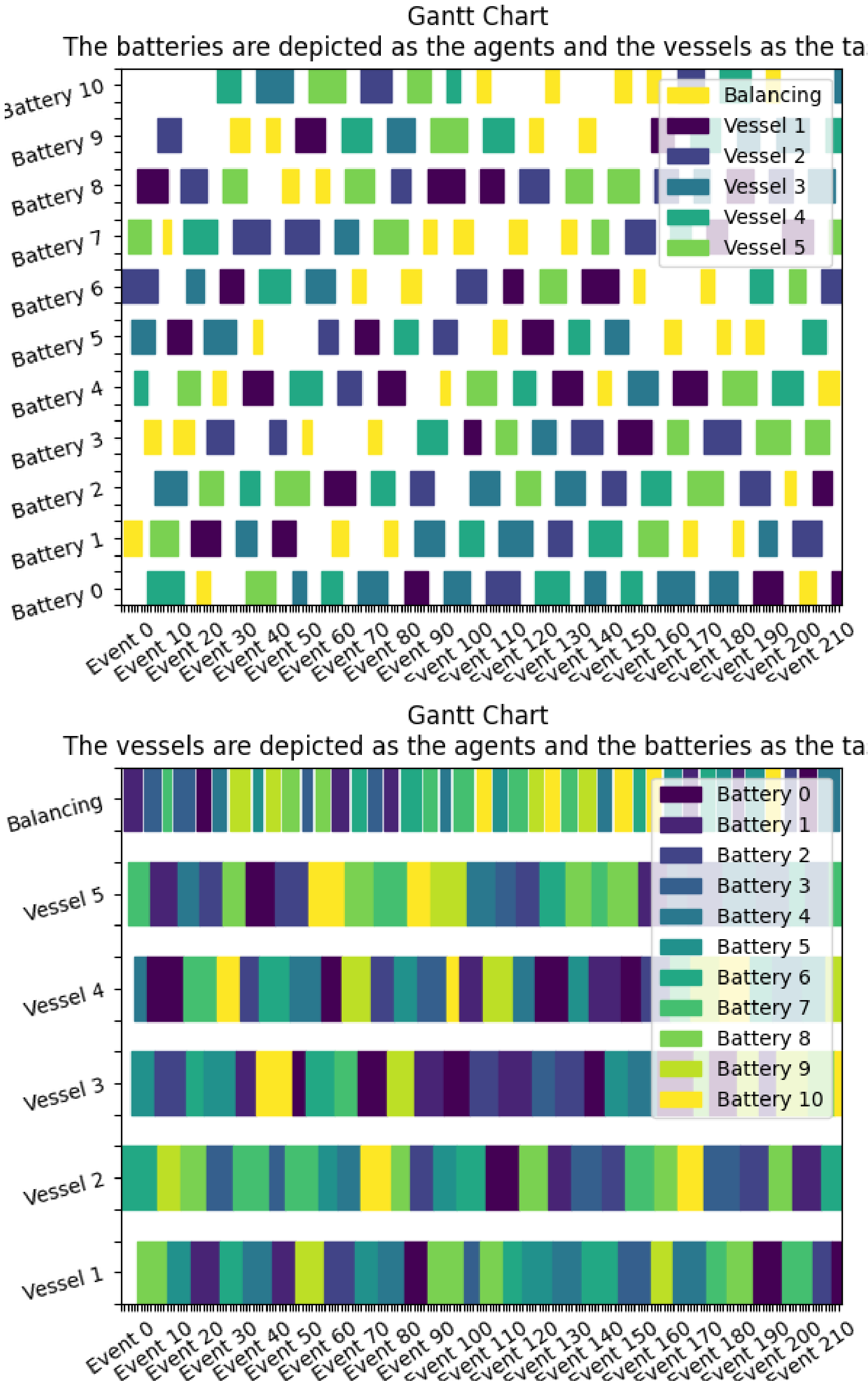


Figure 16: Gantt charts of the solution of the Base Case with all grid balancing stints included using the Approximation Algorithm 3.13. Note that the time axis is event-based rather than time-based. The timeline in Figure 9 presents the event time relation.

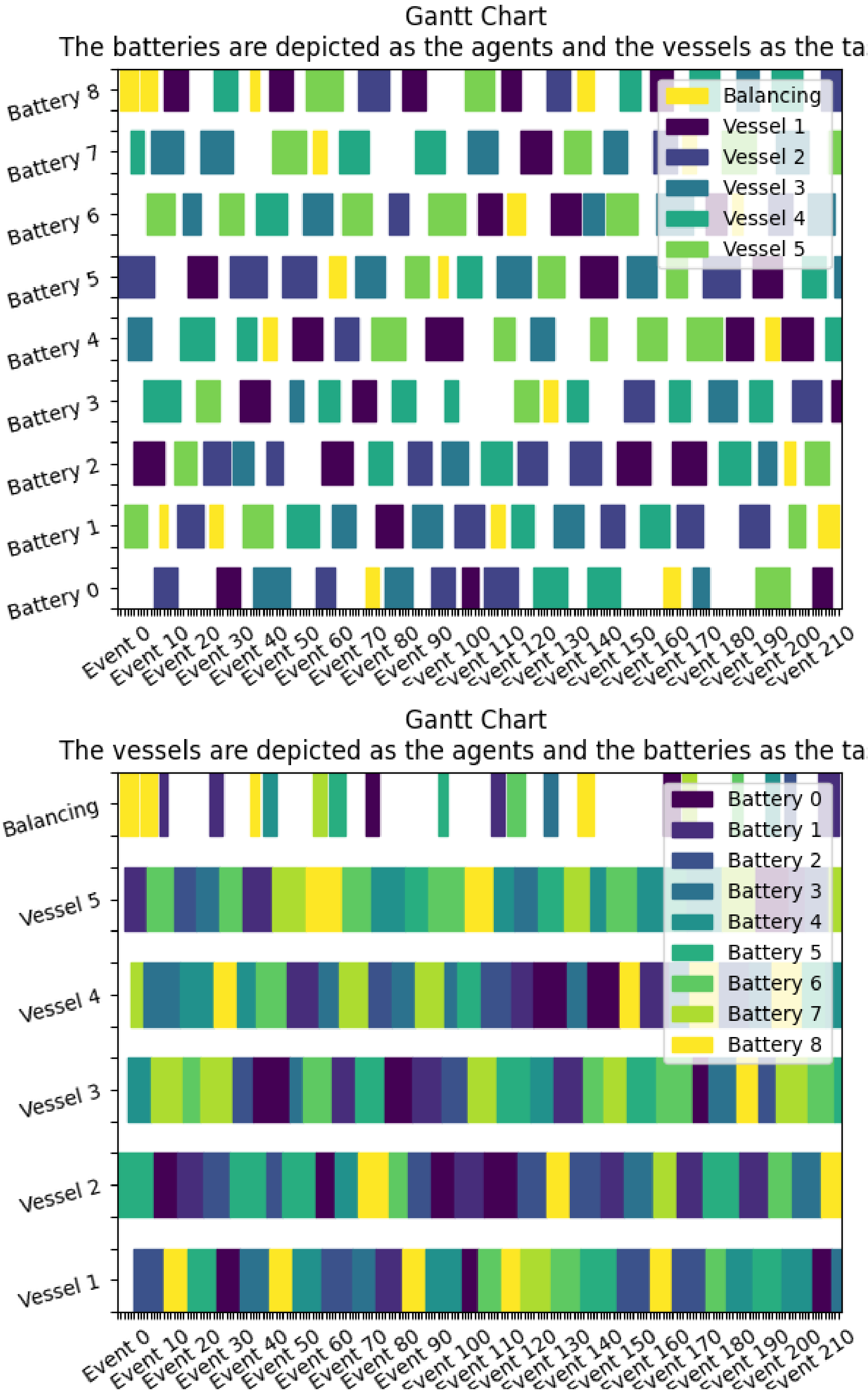


Figure 17: Gantt Charts depicting the results of the Base Case using Revenue Problem Optimization (RPO) (B.6) for the Revenue Problem *RP*. The number of batteries and charging stations is an input found by Battery Priority Optimization (BPO) as shown in Figure 13 and equals $(B, M) = (9, 2)$. Note that the time axis is event-based rather than time-based. The timeline in Figure 9 shows the event time relation.

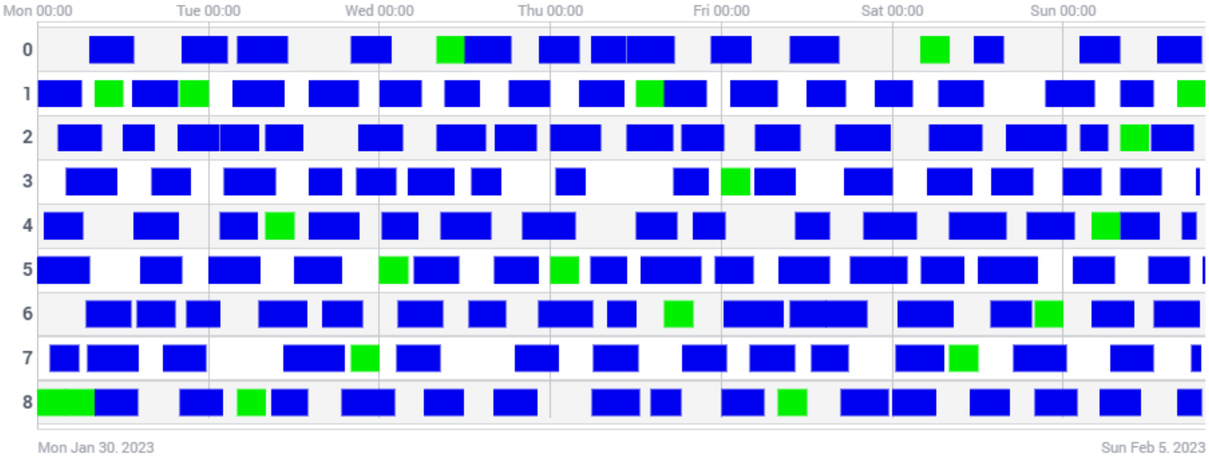


Figure 18: Gantt Charts depicting the results of the Base Case using Revenue Problem Optimization (RPO) (B.6) for the Revenue Problem *RP*. This Gantt chart depicts the same data as Figure 17. However, its x-axis shows the timeline instead of the time events event.

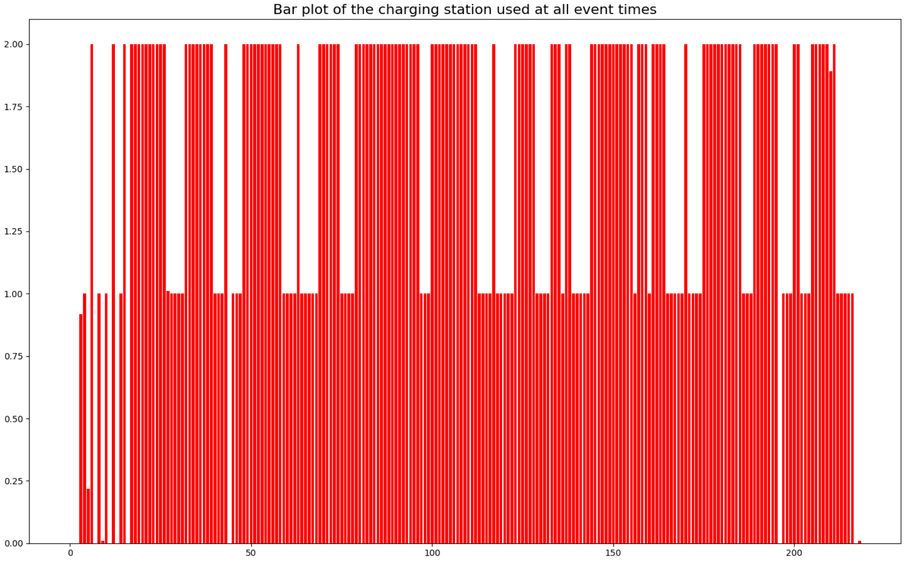


Figure 19: The utilization of charging stations between the time events for the Base Case are computed using the Revenue Problem Optimization. Note that using charging stations for balancing is not included in this figure.

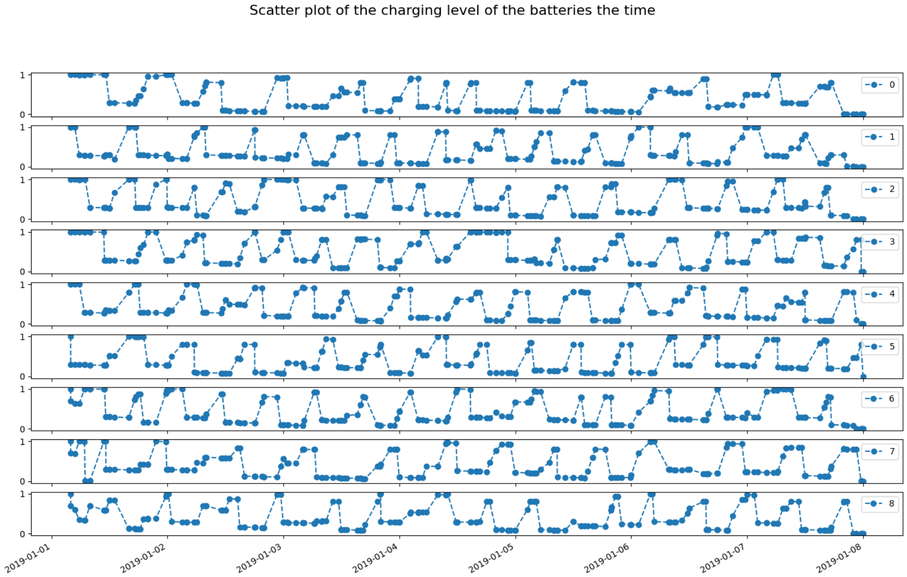


Figure 20: The charging levels of all the batteries for the Base Case are determined using the Revenue Problem Optimization.

6.1.4 Sensitivity Analysis

In Chapter 2, we have defined various input parameters that significantly impact the outcomes of the different approaches described in Section 5.1. Due to time constraints, the runs are limited to a run-time of one hour. All the results that reached the time cutoff are annotated with an asterisk (*) and their optimality gap is stated. Also, to reduce the computational time only the BPO is considered for *FP*, the CSPO takes more batteries into consideration which increases the dimension of the model significantly, as can be seen in Table 7. The input parameters are all analyzed in a similar style. Three types of scenarios are compared: a randomized option, a stylized option, and extreme value cases. The base case scenario is denoted with (*BC*). To analyze the sensitivity of these inputs, the following factors and scenarios are considered:

1. The effects of different types of arrival times (T_a) are analyzed for the following scenarios. The results of *FP* for the following cases can be found in Table 11. The results of *RP* of the same cases are found in Table 12, where different degrees of freedom are explored in the number of batteries used in the system.
    - Random (*BC*): All vessels are simulated independently with a return time normally distributed.
    - Cyclic: All vessels arrive perfectly divided over 6 hours and exactly return 6 hours later.
    - Tight: All vessels arrive 15 minutes apart and return exactly 6 hours later.
2. The effects of different types of energy consumption (E_t) for the different routes in the system. Table 30 in Appendix D shows the inputs of the cases. The results of the following cases can be found in Table 13.
    - Equivalent (*BC*): All routes have the same average energy requirement (see Table 31).
    - Worst case: All routes and balancing stints have the worst energy requirement (see Table 32).
    - Best case: All routes and balancing stints have the best energy requirements (see Table 34).
    - Random: All routes have the randomized energy requirement (see Table 33).
3. The effects of changing the charging speed (d) are analyzed. The results of the following cases can be found in Table 14.
    - Very short: $d = 0.000278$ (charging time equals 1 hour).
    - Short: $d = 0.000139$ (charging time equals 2 hours).
    - Average (*BC*): $d = 0.000100$ (charging time equals 2 hours and 47 min).
    - Long: $d = 0.000069$ (charging time equals 4 hours).
    - Very Long: $d = 0.000046$ (charging time equals 6 hours).
    - Extreme: $d = 0.000035$ (charging time equals 8 hours).
    - No charge: $d = 0$ (infinite charging time), follows from logic as seen in results of the Pareto Front (Equation 40).

4. The effects of the different types of grid balancing length (T_g) and Revenues (R_t) are analyzed. The results for RP of the following cases can be found in Table 15.
    - Equivalent (BC): 4 hours stints and with $R_{t_g} = 1$.
    - Random lengths: simulated similar to the return time of a vessel with an expectation of 4 hours and variance of one hour and with $R_{t_g} = 1$ (see the timing of Table 29).
    - Random Lengths with Scaled Revenue, the revenue of the balancing stints is scaled to their respective lengths (see Table 29 in Appendix D).

The FP results of the different time scenarios, as presented in Table 11, show the impact of the event time-space on the objective functions within a five-vessel single charging port system. The number of batteries required by the system is very dependent on the specific time scenario. The Cyclic case (schedule shown in Figure 43 in Appendix C), where all the vessels arrive perfectly divided throughout the day, shows that the system can be sustained with 8 batteries. However, the Tight scenario (schedule shown in Figure 45 in Appendix C), where not a single battery can be shared amongst the vessels, shows that in some cases 10 batteries are necessary to be able to provide all vessels with sufficiently charged container batteries. On the other hand, the number of charging stations is not very dependent on the time instance of the system. The 2 charging stations produce $2 \cdot d \cdot \Delta_{\text{day}} = 16.52$ batteries worth of charge a day, which is more than enough energy to fully charge all the batteries. All vessels return on average 4 times a day. Therefore, the whole system most likely only needs $5 \cdot 4 \cdot E_t = 14.00$ batteries worth of charge a day. Due to the instant interchanging of the charging batteries, the two stations combined produce enough in all the time instances.

Table 11: Results of Feasibility Problem (FP) for the Deterministic Case of different time scenarios (Point 1 of Section 6.1.4). The results are given in the following format, the methods AA and BPO give the number of battery and charging stations (B, M), where the one generated by BPO is optimal. The optimization methods are listed in Section 6.1.2 with t their respective computational time.

Cases\Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)
Random (BC)	(9, 3)	1.371	(9, 2)	576.516	67.797
Cyclic	(8, 2)	1.295	(8, 2)	3055.719	676.922
Tight	(10, 3)	1.367	(10, 2)	74.984	170.734

The RP results of the different time scenarios, as presented in Table 12, show how many grid balancing spots can be chosen if the optimal number of batteries of FP have been set. Even though the Cyclic case seems to be ideal for the sharing of batteries among the ships, the number of grid balancing slots secured in the case is worse than in the Random case.

The Random case (in column (B, M) of Table 12)) outperforms the Cyclic case (in column $(B + 1, M)$ of Table 12) in revenue with 5 additional grid balancing spots. This is surprising because the Cyclic case has an idle battery in this scenario. As indicated by the RP results of the Base Case, see Table 15, the bottleneck for implementing additional balancing stints lies with the number of charging stations. In the Cyclic case, the constraints imposed by the number of charging stations have a more pronounced impact. This is due to the reduced flexibility in determining the optimal charging times for each battery, leading to a greater revenue increase in the Random Case, despite sharing the same infrastructure. Notably, the Tight case benefits even more flexibility in charging. This allows the Tight case to outperform the Base Case with equivalent infrastructure, see the difference in revenue of (B, M) of Tight and $(B + 1, M)$ of Random in Table 12).

The difference in the balancing stints chosen results from the availability of the charging stations. The Random case has more room to effectively choose what battery to charge to keep a battery aside for balancing, while the Cyclic case is stuck in a fixed cycle of charging that prohibits a charging station to be available for the entire balancing stint.

Table 12: Results of Revenue Problem (RP) for the Deterministic case of different Time cases (Point 1 of Section 6.1.4). (B, M) is the optimum retrieved from Table 11: (9, 2) for Random, (8, 2) for Cyclic, and (10, 2) for Tight. The +1 denotes an extra battery or charging station on top of the previously listed optima.

Cases\Methods	(B, M)	t_{RPO} (s)	Gap %	(B_{+1}, M)	t_{RPO} (s)	(B, M_{+1})	t_{RPO} (s)	Gap %	(B_{+1}, M_{+1})	t_{RPO} (s)
Random (BC)	20	2357.047	0%	21	2068.75	38	248.797	0%	41	14.594
Cyclic	15	859.828	0%	15	277.922	19*	3600.482	5.3%	41	89.719
Tight	27*	3601.953	3.7%	28	54.157	33*	3600.328	3.0%	41	82.319

The energy consumption scenarios show about the same effects on the objective values of FP as depicted in Table 13. A limited effect is expected due to the free nature of the charging choices. There is enough energy added to the system by the charging stations and this charge accommodates all the power needs of the different energy consumption cases. However, energy consumption does have a significant effect on the number of balancing stints chosen. The greater the energy consumption, the fewer balancing stints are employed, as more charging is required primarily to address the departing ships. This leaves insufficient time to accommodate battery recharging in the face of heightened energy consumption associated with a grid balancing stint.

Table 13: Results (FP) Deterministic Case of different energy consumption scenarios (Point 2 of Section 6.1.4). The cases can be seen in Table 30. The results are given in the following format, the methods AA and BPO give the number of battery and charging stations (B, M), where the one generated by BPO is optimal, the RPO method gives the number of additional revenue R that can be generated when using (B, M) generated by BPO. The optimization methods are listed in Section 6.1.2 with t their respective computational time. Note that the RPO of worst case uses an extra battery and charging station compared to the other cases.

Cases\Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)	RPO	t_{RPO} (s)	Gap (%)
Equivalent (BC)	(9, 3)	1.505	(9, 2)	586.578	16.391	20	2357.047	0%
Worst case	(10, 3)	1.545	(10, 3)	1193.609	2556.109	14*	3600.235*	14.3 %
Best case	(9, 3)	1.745	(9, 2)	627.671	3581.343	39	937.672	0%
Random	(10, 3)	1.826	(9, 2)	558.125	52.107	8*	3600.203*	5.9 %

The effect of the charging rate on the infrastructure and the computational time is shown in Table 14. The longer the charging time the more batteries and charging stations are needed. With more batteries to schedule the computational time of the whole system increases. It can be seen that the approximation model (AA) is a good baseline for the number of batteries necessary and can be used to limit the dimension of batteries explored in the specific optimization to reduce the computational time.

Table 14: Results (*FP*) Deterministic Case of charging speeds (Point 3 of Section 6.1.4). The optimization methods are listed in Section 6.1.2 with t their respective computational time.

Cases\Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	Gap	t_{CSO} (s)	Gap	RPO	t_{RPO} (s)	Gap (%)
Very short (1 hour)	(8, 2)	1.253	(9, 1)	611.469	0%	14.031	0%	40	69.782	0%
Short (2 hours)	(9, 3)	1.525	(9, 2)	611.515	0%	16.391	0%	38	574.296	0%
Average (\textit{BC})	(9, 3)	1.505	(9, 2)	586.578	0%	574.391	0%	20	2357.047	0%
Long (4 hours)	(10, 3)	1.829	(10, 3)	920.063	0%	113.281	0%	31*	3600.719*	4.8%
Very Long (6 hours)	(13, 5)	1.505	(10, 4)	1963.641	0%	947.797	0%	23*	3600.875*	11%
Extreme (8 hours)	(15, 6)	3.155	(12*, 5)	3600.250*	16.7%	3060.546	0%	19*	3600.328*	115.8%
No-Charge	(137, ∞)	22.865	-	-	-	-	-	-	-	-

In reality, the grid balancing stints might not be a fixed length. Therefore, the Base Case is also compared to a case with random lengths of stints. Additionally, to not cherry-pick all the short balancing stints another case is added in the comparison that scales the revenue per stint based on its length. Table 15 shows the results of this comparison. As expected, the random lengths grid balancing case outperforms the Base Case in revenue. However, it does choose the shorter balancing spots. This can be seen by the fact that the total duration of all stints decreases while the revenue increases. For the scaled revenue case can be seen that the total grid balancing length is larger than the Base Case. Different options of lengths are likely beneficial for generating revenue.

Table 15: Results (*FP*) Deterministic Case of different lengths of grid balancing stints (Point 4 of Section 6.1.4). RPO is the Revenue problem optimization, where the number of batteries and charging stations is equal to the $(9, 2)$ that are found for the Base Case. The length is the tally of the balancing stints chosen by the optimization.

Cases\Methods	RPO	t_{RPO} (s)	Gap	total length
Equivalent (*BC*)	20	2357.047	0%	3 days, 8:00:00
Random Lengths	22*	3600.341*	6.2%	3 days, 7:49:56
Random Lengths with Scaled Revenue	20.733*	3600.284*	5.2%	3 days, 10:56:08

6.1.5 Overlapping Grid Balancing Analysis

The optimizations have been conducted under the assumption that a single grid balancing spot is available at all times. However, some additional space is observed in the optimal RP schedule found for the Base Case. In reality, multiple revenue options may occur simultaneously, for example, if multiple energy companies have different slots for grid balancing available. Therefore, scenarios are explored with two grid balancing stints that are active simultaneously. The following two scenarios are explored.

1. Overlapping: grid balancing stints start every two hours and have a 4-hour duration.
2. Two Random: two grid balancing contracts are randomly simulated as done for vessel arrival time with the duration of the stints expected to be four hours with a variance of an hour. The revenue is adjusted to the respective duration.

The results shown in Table 16 show that more grid balancing stints can be chosen. The ability for the system to cherry-pick the grid balancing stints enables more revenue to be gained. The results outperform the results of adding additional batteries or a charging station to the system. This result is significant because it shows that to increase the revenue it can be more profitable to increase the revenue options than to add an extra battery or charging station to the system.

Figure 21 and Figure 22 show example schedules for both cases that accomplish the revenues denoted in Table 16. The idle spaces in the Gantt chart are filled in more and overall the batteries are used more optimally. There even seems to be more white space that could be filled with balancing stints tailored to the case at hand.

Table 16: Results (RP) Deterministic Case of the overlapping grid balancing scenarios. RPO is the Revenue problem optimization, where the number of batteries and charging stations found for the Base Case $(9, 2)$ is used as the input and the number of accepted grid balancing stints is optimized.

Cases\Methods	RPO	t_{RPO} (s)	Gap	length
Base Case	20	2357.047	0%	3 days, 8:00:00
Overlapping	24*	3600.688	4.5%	4 days
Two Random	26.350*	3600.390	8.0%	4 days, 9:23:54

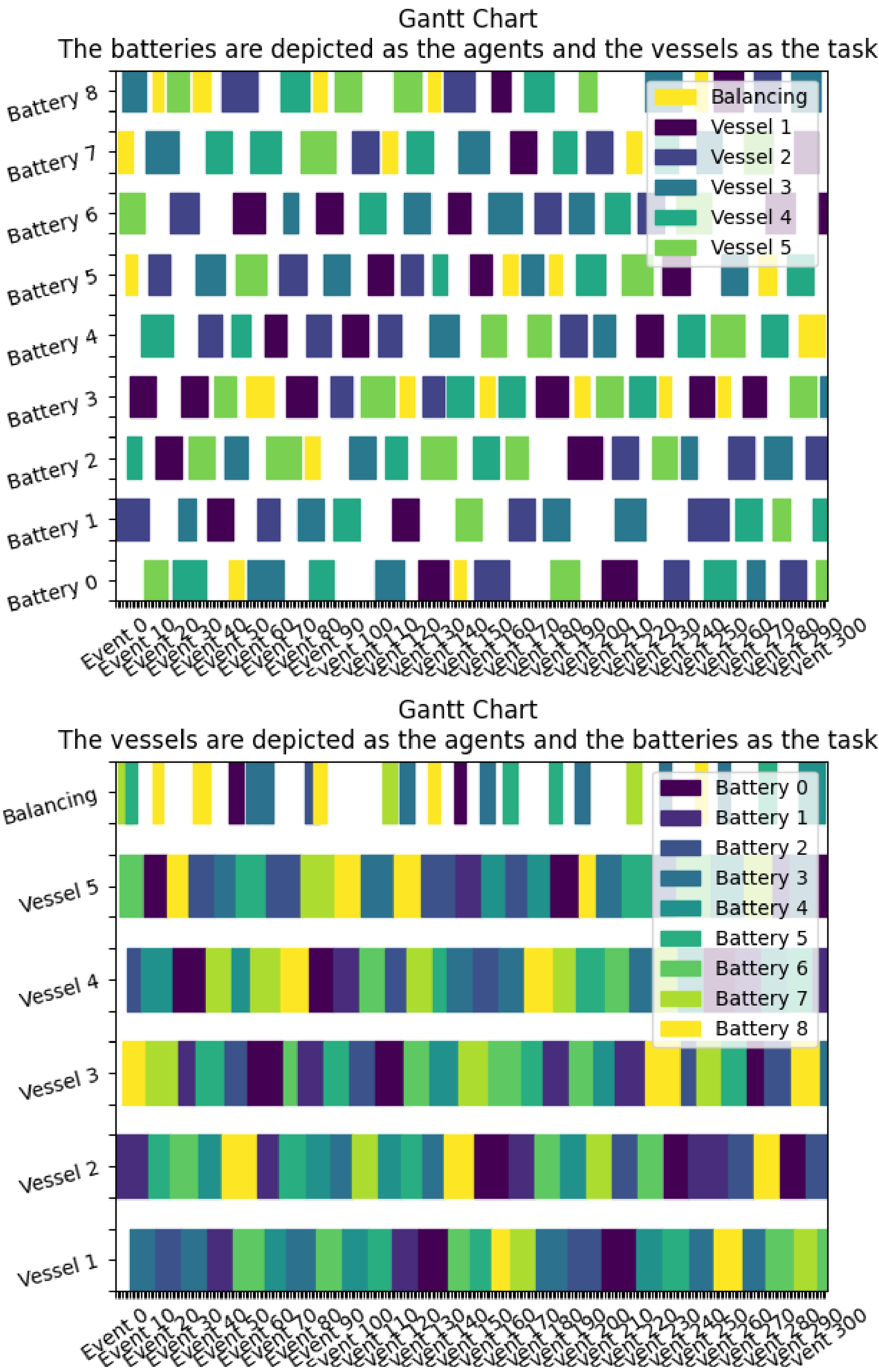


Figure 21: Gantt Charts depicting RPO outcome for the data case with overlapping grid balancing stints with fixed lengths of 4 hours with starting times 2 hours apart.

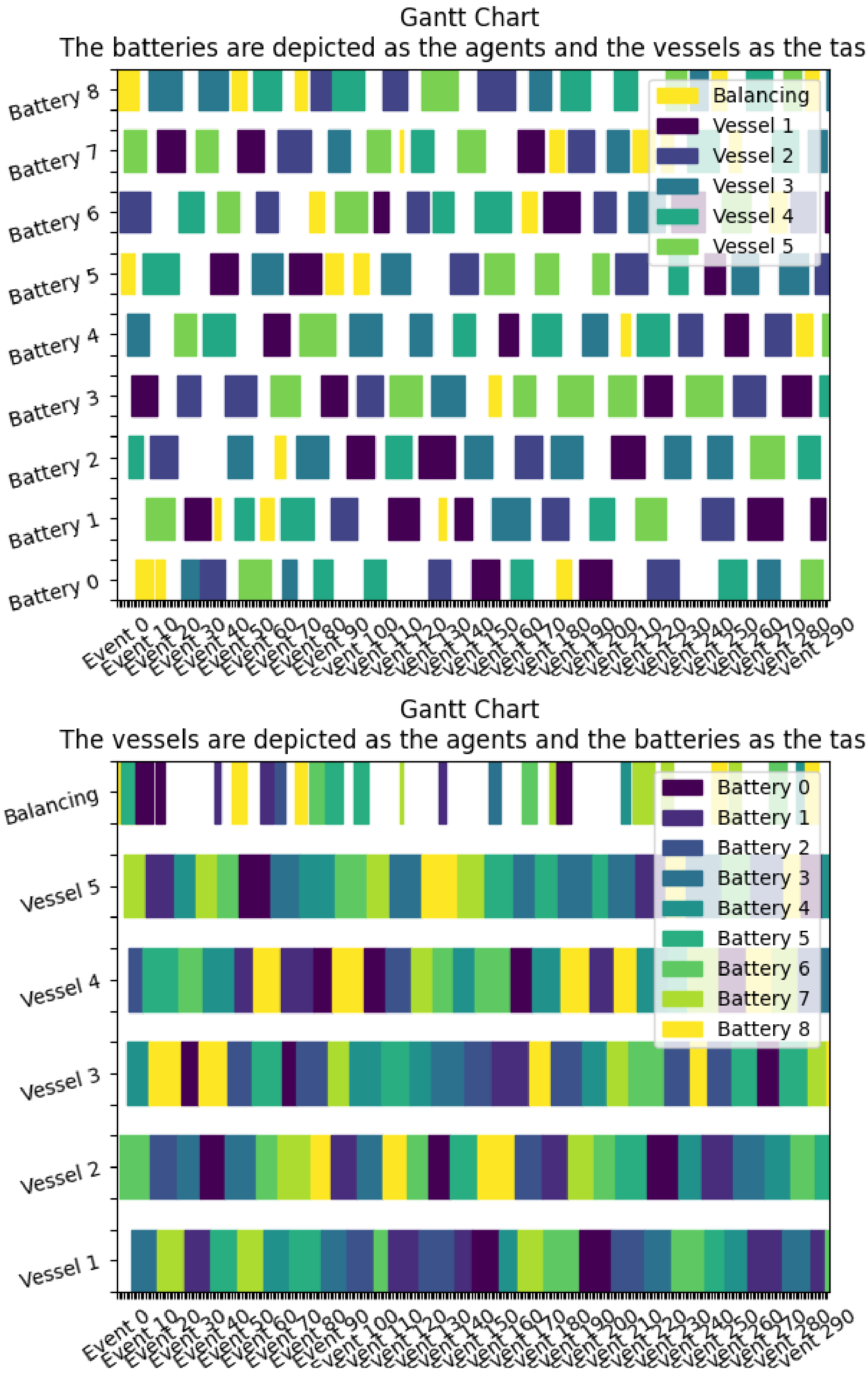


Figure 22: Gantt Charts depicting the outcome of the case with randomly generated grid balancing stints that overlap. The timing and duration of the tasks that follow from the simulation can be found in Table 29.

6.1.6 Computational Analysis

To investigate the scalability of the deterministic model the computational time of different sizes of the model is compared. The main dimension that the problem scales in is the event time-space of the model. Scaling of the time dimension can be done in two manners:

1. increasing the length of the time horizon
2. increasing the number of events on the current horizon.

The results are given in the following format: the methods AA and BPO give the number of battery and charging stations (B, M), where the one generated by BPO is optimal, the RPO method gives the number of additional revenue R that can be generated when using (B, M) generated by BPO. Figure 47 in appendix C shows the schedules generated by the approximation algorithm for different time horizons, BPO and RPO produce similar results. The optimization methods are listed in Section 6.1.2 with t their respective computational time.

Table 17 presents the impact of an increasing time horizon, while Figure 23 illustrates the corresponding computational times. The relationship between fleet size and computational time is described in Table 18, and Figure 24 visualizes this relation. Both figures clearly indicate an exponential growth in the computational time given an increase in input size for the MIP-based optimizations (BO, RPO). However, CSO does not follow this trend due to its reliance on BO's optimal solution as a starting point and therefore intrinsically using its computations. If the CSO is the only optimization run its running time also shows an exponential growth. Therefore, this result is skewed. The exponential growth of all the optimizations resonates with the $\mathcal{NP}$-hardness of the problems.

The objective of the RPO in Table 24 illustrates the effect that both the number of batteries as well as charging stations have on the amount of grid balancing stints that can be chosen. As the number of spare batteries onshore increases, there is a greater chance of having one available for grid balancing. However, when more batteries have to share a limited number of charging stations, the available time to utilize these stations for balancing decreases. The optimal number of grid balancing stints increases drastically if the ratio of charging stations to container batteries rises.

Table 17: Results (FP) Deterministic Case of different time horizons. The optimization methods are listed in Section 6.1.2 with t their respective computational time.

Days\Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)	RPO	t_{RPO} (s)
1	(9, 2)	0.186	(9, 2)	4.719	14.109	5	32.922
2	(9, 2)	0.392	(9, 2)	5.859	7.328	8	133.047
3	(9, 3)	0.619	(9, 2)	86.969	35.391	10	584.75
5	(9, 3)	1.096	(9, 2)	226.266	19.562	15	1210.406
7 (BC)	(9, 3)	1.371	(9, 2)	586.578	16.391	20	2357.047

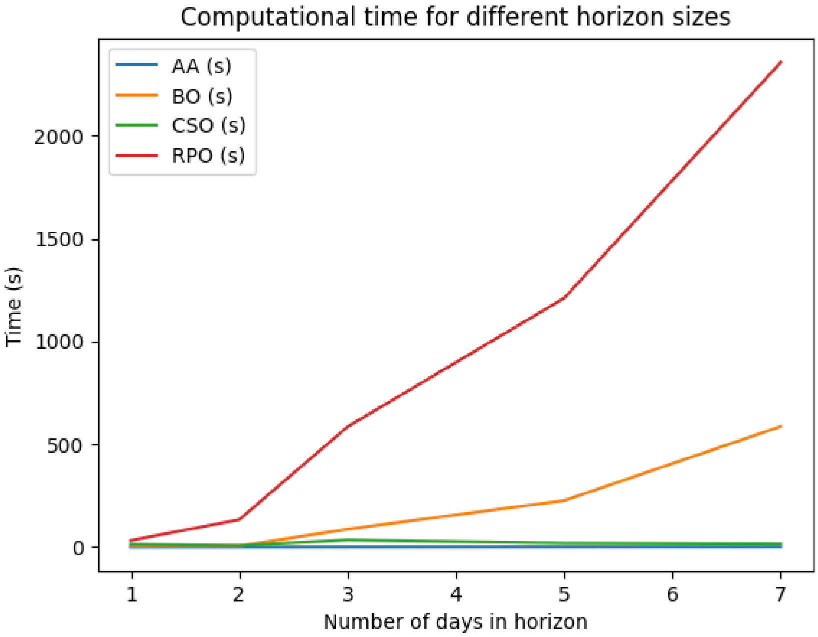


Figure 23: The computational time of different time horizons (1, 2, 5, 7 days) in the deterministic case for different optimizations.

Table 18: Results (FP) deterministic case with different fleet sizes. The results are given in the following format, the methods AA and BPO give the number of battery and charging stations (B, M), where the one generated by BPO is optimal, the RPO method gives the number of additional revenue R that can be generated when using (B, M) generated by BPO. Figure 47 shows an example of the schedules generated by the approximation algorithm. The optimization methods are listed in Section 6.1.2 with t their respective computational time.

Vessels \ Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)	RPO	t_{RPO} (s)
1	(2, 1)	0.081	(2, 1)	0.250	0.110	1	0.156
2	(4, 2)	0.246	(4, 1)	0.843	0.547	19	47.39
3	(6, 2)	0.669	(6, 1)	74.672	2.531	6	93.297
4	(8, 2)	1.232	(8, 2)	119.50	19.687	36	1411.447
5 (BC)	(9, 3)	1.371	(9, 2)	586.578	16.391	20	2357.047
6	(11,3)	3.437	(11, 2)	2093.125	103.062	4	25201.125

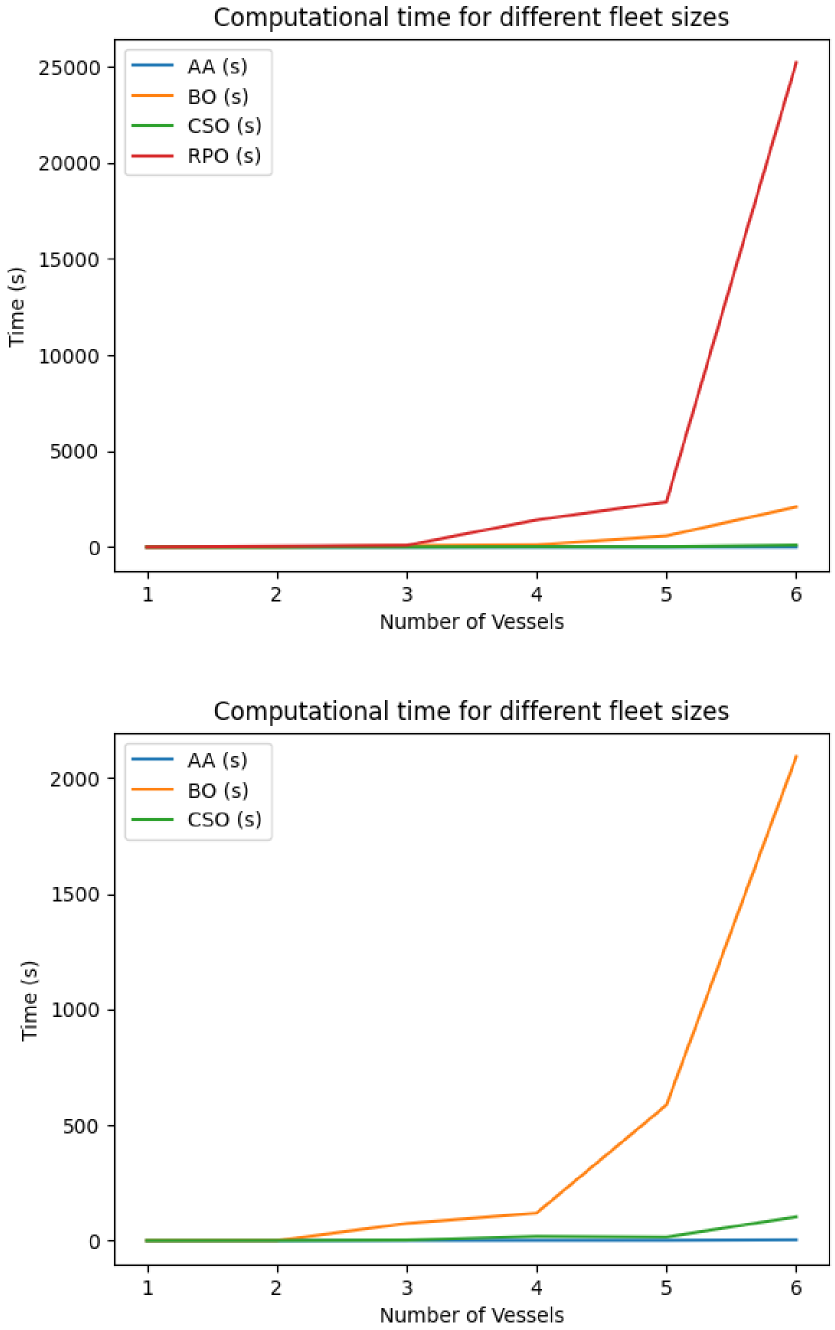


Figure 24: The computational time of different fleet sizes in the deterministic case for different optimizations. The bottom plot shows the same figure but without RPO to show the trend of the other optimizations.

6.2 Case 2: Energy Uncertain Case

For the second case, we assume the energy consumption (E_t) to be uncertain. From Table 33, in the sensitivity analysis done for the deterministic case (Section 6.1.4), we can conclude that the number of batteries under energy consumption uncertainty is likely to remain at 9 or 10 and the number of charging stations at 2. Therefore, this case assumes that the batteries and charging stations are known. This case fully focuses on maximizing the revenue objective (RP).

The energy consumption E_t is a value in the interval $[-1, 1]$ where 1 would mean that 100% of the container battery capacity is used on the task starting at t and -1 would mean that 100% of the capacity must be added to the charging level. However, we choose to assume that vessels cannot add energy to the batteries, uses a least 50% of the battery capacity and never uses the full capacity of the battery, at most 90%. Also, we assume that grid balancing options do not add or take the full capacity of the battery. A battery that is allocated to grid balancing could potentially gain charge. These assumptions are combined in the following assumption that creates the uncertainty sets for all the energy consumption:

Assumption 6.2 (Energy consumption uncertainty sets)**.** Dependent on the type of event t, all the time-events $t \in \mathcal{T}$ have an energy consumption within one of the following sets:

$$E_{t_a} \in [0.5, 0.9]$$

$$E_{t_g} \in [-0.4, 0.6]$$

The different methods used to analyze this case use different types of distributions on the uncertainty sets. To avoid infeasibility in optimization the $C_{t_a}^-$ is adjusted to be 90 %, otherwise values for E_t larger than 0.8 may lead to a negative charge level which violates its range constraints.

6.2.1 Base Case 2

Consider the Base Case defined in Section 6.1.1. Base Case 2 is equivalent to the Base Case defined in Section 6.1.1 except for the values of E_t and C_t^-. The values for E_t are simulated from a uniform distribution over the uncertainty sets shown in Assumption 6.2. Table 36 in Appendix D shows a simulation of this case. For all arrival times, the feasible charge interval becomes $C_{t_a} = [0.9, 1.0]$. The interval C_{t_g} is not changed, however, if the sample E_{t_g} is not viable within C_{t_g} the battery will be returned with either a full charge or an empty charge dependent on the overshoot or undershoot. Table 19 shows the results of this simulated case using the deterministic methods listed in 6.1.2. The gap in RPO shows that the system cannot prove in reasonable time that a schedule with 9 batteries and 2 changing stations does not exist with 19 balancing options. Figure 26 shows the respective schedule found.

Table 19: Results simulated version of the Energy Uncertain Case treated as a deterministic case with simulated values listed in Table 36.

Case \ Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)	RPO (9, 2)	t_{RPO} (s)	Gap (%)	RPO (10, 2)	t_{RPO} (s)	Gap (%)
Base Case 2	(9, 3)	2.249	(9, 2)	3224.187	2764.953	18*	3600.142	11.1%	20*	3600.423	7.6%

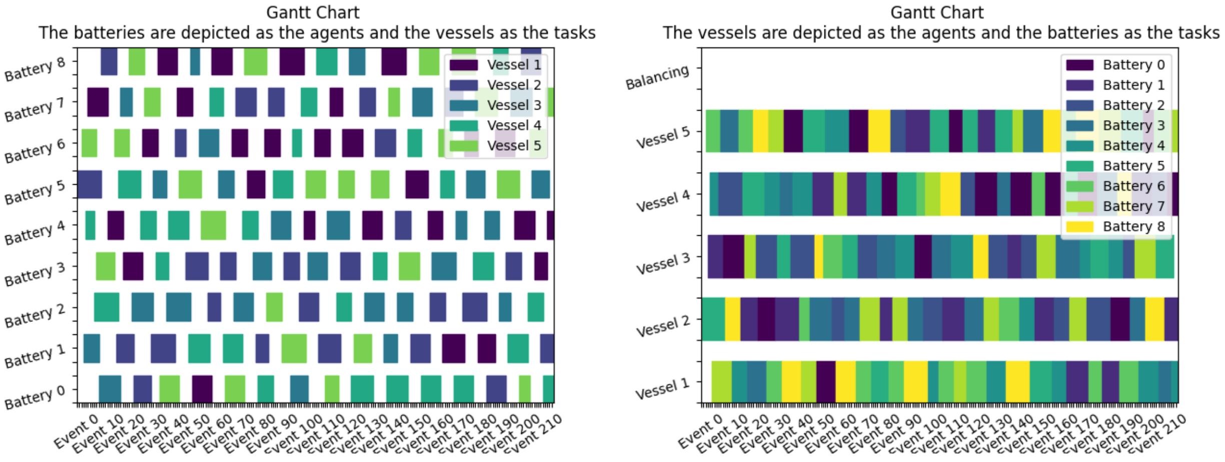


Figure 25: Gantt Charts depicting AA outcome for Base Case 2

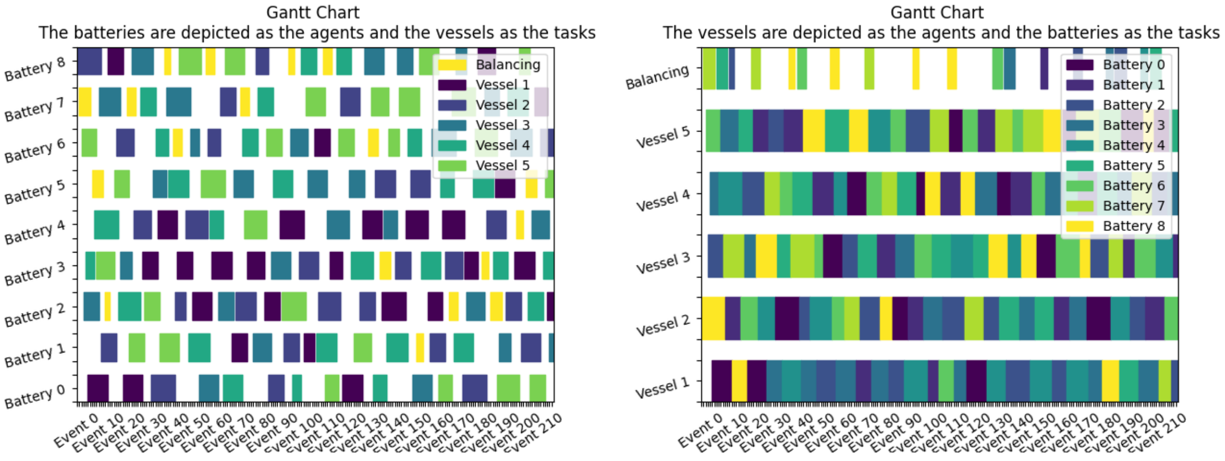


Figure 26: Gantt Charts depicting RPO outcome for Base Case 2 with 9 batteries and 2 charging stations

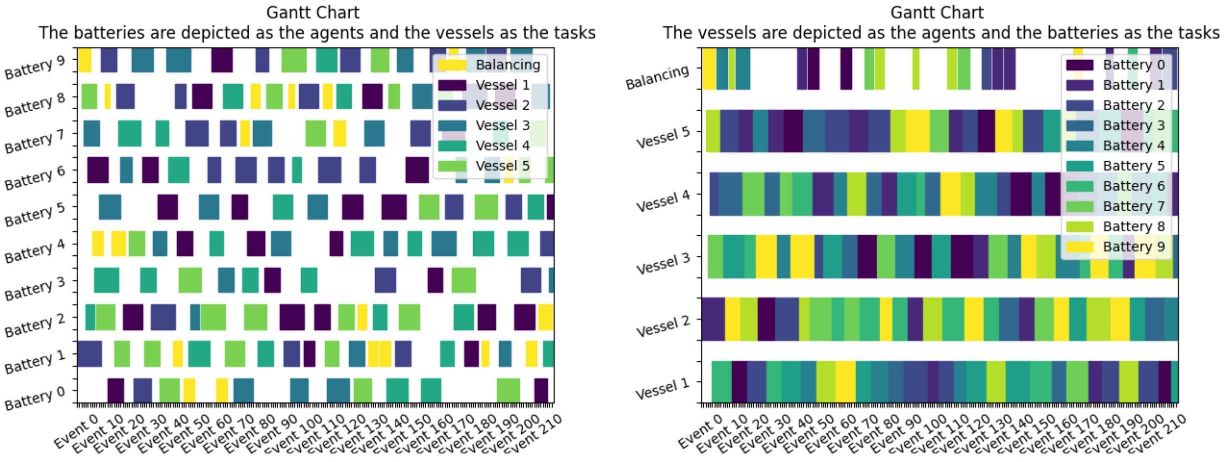


Figure 27: Gantt Charts depicting RPO outcome for Base Case 2 with 10 batteries and 2 charging stations

6.2.2 Optimization Strategies

To include the energy consumption uncertainty, different methods of optimization are used that use different assumptions on the uncertainty sets of Assumptions 6.2. Because of the extended computational times observed in the previous case, the direct Monte Carlo simulation (DCMS) is excluded as an optimization strategy. This method requires a substantial number of discrete optimizations to be performed, which becomes impractical due to the time a single optimization takes.

1. Rolling horizon analysis (RH) as described in Section 5.2.2, assumes a uniform distribution on the uncertainty sets.

   $$E_{t_a} \sim \mathcal{U}([0.5, 0.9]), E_{t_g} \sim \mathcal{U}([-0.4, 0.6])$$

   The rolling horizon with Monte Carlo sampling within this uncertainty set is equivalent to Base Case 2. The rolling horizon takes multiple horizons of 24 hours and iteratively shifts the horizon for 6 hours further. The total number of optimizations ran equates to 24 runs of 24 hours to cover the full week.

2. Extreme case analysis as described in Section 5.3.1 is already showcased in the deterministic case, see Section 6.1.4. This consists of Worst case energy consumption (Worst case) and Best case energy consumption (Best case) optimization. Table 30 in Appendix C shows the values implemented for E_t.

3. Probabilistic constraint analysis (PC) as described in Section 5.2.3 assumes a value for the revenue and optimizes the schedule to minimize the probability of failure at every time event. The probabilistic constraint approach assumes that

   $$E_{t_a} \sim \mathcal{N}(0.7, \sigma_a^2), E_{t_g} \sim \mathcal{N}(0.1, \sigma_g^2)$$

   with the (100 - α)% confidence interval lying within the uncertainty sets. Assume that $\alpha = 1\%$, then we find the variance of the distributions by constructing a 99% confidence interval within the uncertainty set described in Assumption 6.2. The z-score of the confidence interval of the standard normal distribution with $\alpha = 0.01$ equals 2.576. This results in the following variances:

   $$\sigma_a^2 = (0.200/2.576)^2 = 0.0776^2 = 0.0060$$

   and

   $$\sigma_g^2 = (0.500/2.576)^2 = 0.1941^2 = 0.0378.$$

Rolling horizon employs the same optimization method for every horizon, however the starting conditions switch between all horizon optimization. The difference in expected and realized energy consumption is compensated in between the horizon optimization which results in a sudden shift in charge level. This shift may ensue an infeasible time arrival at the beginning of the following horizon. To combat this, a charging level violation (V_b^t) is added to Constraint 7 and 10. This violation is very largely penalized in the objective. The MIP formulation is shown in Appendix B.7. The violation makes sure that the system can keep on running even if the infeasible intermediate states show up. The violation is reported in the percentage of the capacity of the battery added to the system. The time that it would take the system to gain this added energy can be calculated by using the charging speed.

Extreme case analysis uses the revenue optimization problem (RPO) as its solution method, which is equivalent to the deterministic method used for the Base Case. The corresponding MIP formulation is shown in Appendix B.6. Probabilistic constraint analysis has its separate MIP formulation, which can be found in Appendix B.8. The revenue values tested in the probabilistic constraint method are those falling within the range of results determined by the extreme case analysis.

6.2.3 Results

The rolling horizon analysis is compared to the results of the Base Case. Subsequently, the extreme case analysis shows the limits of the influence of energy consumption uncertainty. The probabilistic constraint approach is employed to determine the most robust schedules for revenues situated within the boundaries established by the extreme case analysis.

Rolling horizon is implemented in the following two cases.

1. No Pull Case (NPC), there is no Monte Carlo simulation in between horizon optimizations.
2. Random Pull Case (RPC), the intermediate values are randomly simulated from the uncertainty set.

The simulations are ultimately fed into a deterministic *RPO* optimization. Then, we are able to compare the rolling horizon version to its retrospective deterministic counterpart. Table 20 shows the result of the rolling horizon test cases.

Optimization\ Method	RH	t_{RH} (s)	Violation (%)	RPO	t_{RPO}	Gap (%)
NPC	17	733.730	0%	21	2068.750	0%
RPC	15	1261.220	14.4%	20*	3600.423	7.6%

Table 20: Results of the rolling horizon method on the case with (NPC) and without Monte Carlo simulation (RPC) compared to their retrospective deterministic counterparts. The infrastructure is set to 10 batteries and 2 charging stations. The violation found in RPC equates to around 22 min of waiting time across 6 time arrival events.

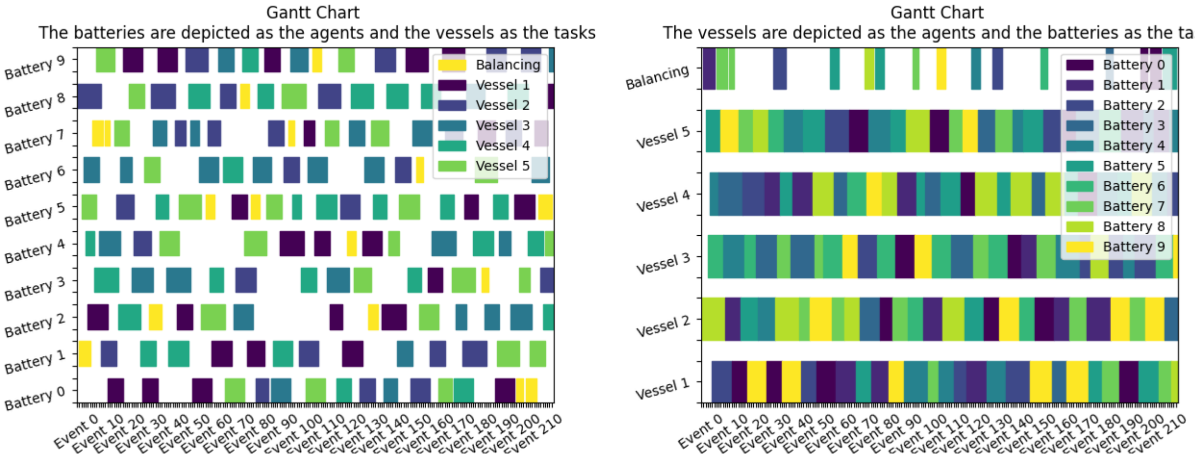


Figure 28: Gantt charts of the rolling horizon without Monte Carlo simulation (NPC). The retrospective counterpart of this case corresponds to the Base Case, defined in Section 6.1.1.

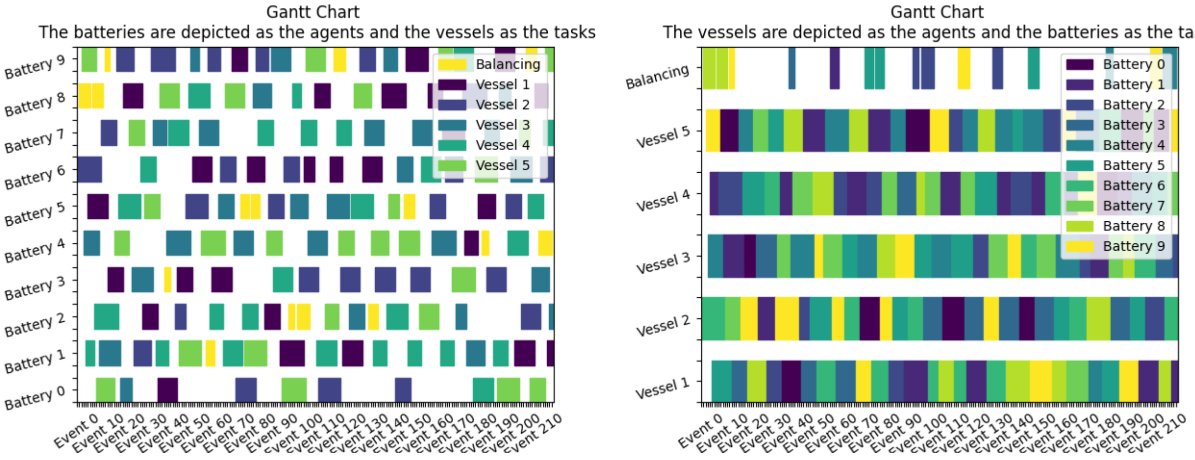


Figure 29: Gantt charts of the rolling horizon with Monte Carlo simulation (RPC). The retrospective counterpart of this case corresponds to Base Case 2, defined in Section 6.2.1.

The extreme case analysis is performed in the sensitivity analysis of deterministic case (Section 6.1.4). The worst case is only feasible for the system with 10 batteries and 3 charging stations. Figures 30 and 31 show the schedules of the extreme cases where both 10 batteries and 3 charging stations are used, where the base case makes use of all the grid balancing spots due to the idle battery and charging station compared to its BPO produce schedule. These results show the impact the energy consumption uncertainty may produce. While these particular instances are improbable, they establish the upper and lower boundaries for attainable revenue within the system.

Table 21: Recap of energy consumption extreme case revenue solutions as seen in Table 30. Note that worst case is optimized with infrastructure (10, 3) and best case with (9,2).

Cases\Methods	RPO	t_{RPO} (s)	Gap (%)
Worst case	14*	3600.235*	14.3 %
Best case	39	937.672	0%

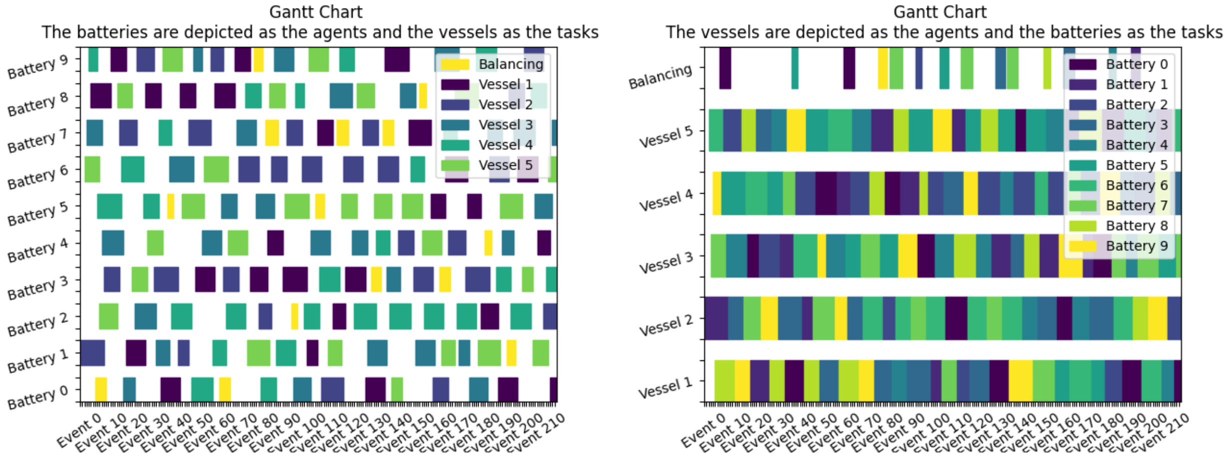


Figure 30: Gannt chart of RPO results worst case energy consumption scenario with 10 batteries and 3 charging stations.

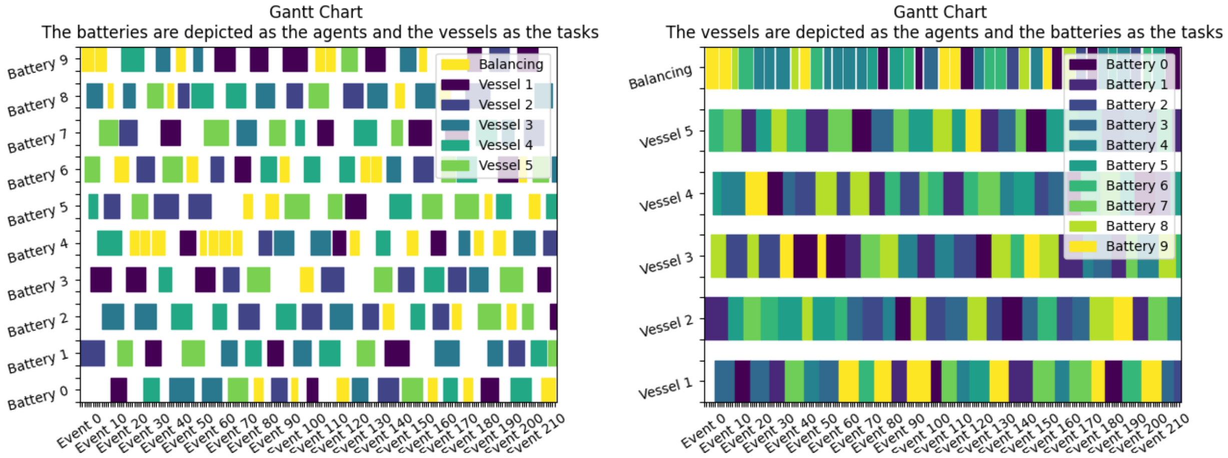


Figure 31: Gannt charts of RPO results best case energy consumption scenario with 10 batteries and 3 charging stations.

The probabilistic constraint approach can be employed on the revenues that lay within the worst and best-case values. We assume that 10 batteries and 2 charging stations are employed in the system. For the chosen uncertainty sets in this case, every revenue from 0 to 40 lies within the values found for extreme cases. To limit the experiment time all optimizations are limited to 10 minutes.

Table 22 shows the results of the probabilistic constraint method applied to all revenue values. The system failed to find a solution within 10 minutes for a revenue greater than 21. The table clearly shows an increasing probability of failure at higher revenues gained. This correlates with the expectancy that accepting additional tasks increases the probability of failure.

Figure 32 shows the schedule found for revenue of 21. Under Assumption the assumption that the energy consumption is normally distributed over the uncertainty set defined in Assumption 6.2, the maximum probability no battery with sufficient charge is ready at the harbor equals 44.3% at time event 152. The system ensures that all other time events have a lower chance of failure.

Table 22: Results of the probabilistic constraint for revenue values within the extreme scenarios methods with a time limit set to 5 minutes. The critical value β is optimized by probabilistic constraint. If a sample is drawn from the standard normal distribution at time event t and exceeds β, it leads to a scheduling failure. This β is larger or equal to all critical values at other time events.

Revenue	β	$P(z > \beta)$	t_{PC} (s)	Gap (%)	Time event t
0	1.1887795	0.117263230	600.235	2.05424	070
1	1.1369536	0.127778834	600.421	2.06867	057
2	1.0877297	0.138357226	600.250	1.69011	040
3	1.0346558	0.150414836	600.391	1.73702	067
4	0.9772088	0.164232893	600.375	1.80175	029
5	0.9214808	0.178399735	600.469	1.87355	028
6	0.8640441	0.193781837	600.187	2.05015	044
7	0.8081879	0.209491213	600.437	2.25594	055
8	0.7562307	0.224755435	600.562	2.15074	019
9	0.7023791	0.241221379	600.437	2.43661	038
10	0.6548198	0.256291894	600.359	2.14099	028
11	0.6066775	0.272032494	600.500	2.25537	036
12	0.5605540	0.287550799	600.390	2.38706	036
13	0.5157684	0.303008074	600.422	2.63507	040
14	0.4726980	0.318214332	600.375	2.85265	037
15	0.4326387	0.332638639	600.328	2.77094	142
16	0.3890788	0.348608936	600.328	3.93701	023
17	0.3381372	0.367629906	600.375	7.69287	105
18	0.3159114	0.376034880	600.407	2.98269	076
19	0.2735699	0.392207586	600.391	5.21358	178
20	0.2385810	0.405715279	600.281	5.29492	147
21	0.1431082	0.443102351	600.297	50.78289	152
22	infeasible	1.00	600.296	-	-

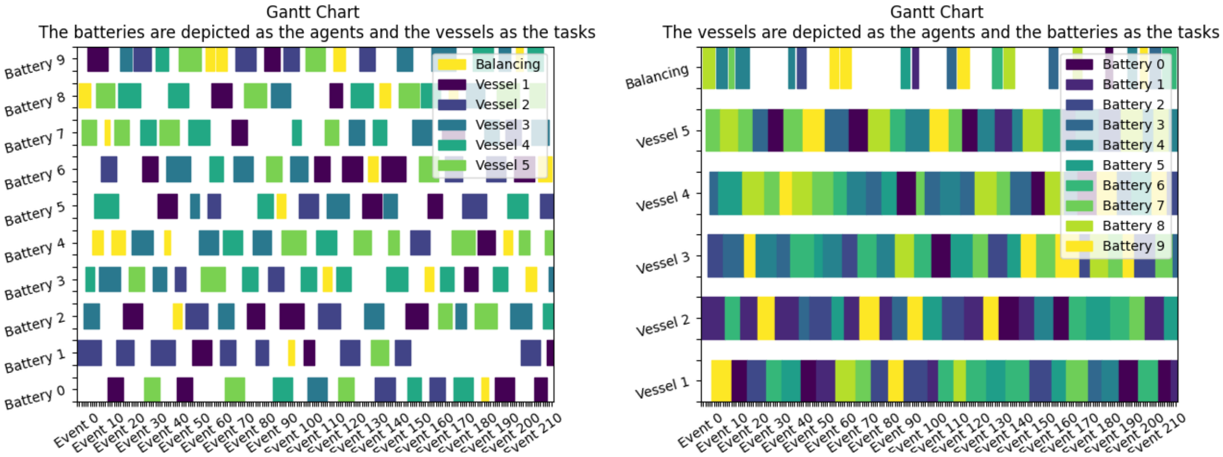


Figure 32: Gannt charts of PC results with the scenario of 10 batteries 2 charging stations and 21 revenue.

6.3 Case 3: Time Uncertain Case

For the second case, we assume the arrival times (t_a) to be uncertain. The sensitivity analysis (see Section 6.1.4) indicates that certain time scenarios necessitate 10 batteries and 2 charging stations. Due to the small effect on the batteries and charging stations given different time cases, we proceed with the assumption that this case will have 10 batteries and 2 charging stations and focus on the changes in revenue given time arrival uncertainty.

There are three kinds of uncertainty sets considered for the arrival times.

1. All the arrival times are pulled independently from a fixed time frame.
2. All the arrival times are pulled independently from a dynamic time frame.
3. Only the first arrival times of the ships are pulled independently from a dynamic time frame.

The fixed time frame is assumed to be an hour on both sides of the expected arrival time, the dynamic time frame is 10% of the duration on both sides of a time frame. The dynamic time frame incorporates that as the expected arrival time draws closer, the uncertainty interval narrows, whereas the fixed time frame stays the same. The dynamic time frame is closer to reality due to the constantly updated eta predictions of the vessels.

Respectively the different types of uncertainty sets generate the following three assumptions.

Assumption 6.3 (Fixed 1 hour distribution)**.** All the arrival times are independently uniformly distributed around the expected arrival time values $\bar{T_a}$ listed in the Base Case.

$$T_a \in [\bar{T_a} - 1 \text{ hour}, \bar{T_a} + 1 \text{ hour}]$$

Assumption 6.4 (Individually Independently Distributed (IID) Arrival Time Uncertainty Sets)**.** All the arrival times are independently uniformly distributed around the expected arrival times $\bar{T_a}$ listed in the Base Case (26). Now we assume an uncertainty of 10%. This gives the following uncertainty set:

$$T_a \in [\bar{T_a} - 0.1 \sum_{t \leq t_a} \Delta_t, \bar{T_a} + 0.1 \sum_{t \leq t_a} \Delta_t]$$

Assumption 6.5 (Vessel Dependently Distributed (VDD) Arrival Time Uncertainty Sets)**.** Every vessel is independently uniformly distributed around the values listed in the Base Case, the other arrivals times of the vessel are scaled with the same pull.

$$U_s \in [0.1 \sum_{t \leq t_{s_1}} \Delta_t, 0.1 \sum_{t \leq t_{s_1}} \Delta_t]$$

where t_{s_1} is first arrival time of vessel s. Subsequently, all arrival times are shifted with U_s, i.e. $T_{a_s} = \bar{T}_{a_s} + U_s$ where a_s is an arrival time of ship s.

6.3.1 Base Case 3

The Base Case defined in Section 6.1.1 is also used for the Base Case of this case. The results for this case can be found in Section 6.1.3. Table 23 shows a short summary of the results.

Table 23: Resuls of Base Case as seen in Tables 9 and 10 in Section 6.1.3. Note that RPO is performed with 10 batteries and 2 charging stations.

Case \ Methods	AA	t_{AA} (s)	BPO	t_{BO} (s)	t_{CSO} (s)	RPO (10,2)	t_{RPO} (s)
Base Case	(9, 3)	1.371	(9, 2)	576.516	67.797	21	2068.750

6.3.2 Optimization Strategies

The following two optimization methods are used for the time uncertainty case.

1. Rolling horizon analysis (RH), the Base Case is tested and additionally two scenarios are simulated, one under Assumption 6.4 and the other under Assumption 6.5. The optimizations are carried out as explained in Section 6.2.2. The MIP formulation can be found in Appendix B.7. Both simulated timelines are fed into a deterministic RPO optimization to show their relative performance compared to the deterministic case, where all the information is available.

2. Value at risk analysis (WC). The arrival times are split as explained in Section 5.3.2. The set splitting uses the uncertainty set defined in Assumption 6.3. Figure 38 shows the split timeline of the first day.

6.3.3 Results

Rolling horizon is tested in the following scenarios.

1. No Pull Case (NPC), there is no Monte Carlo simulation in between horizon optimizations.

2. Individually Independent Pull Case (IIPC), the intermediate values are simulated from the uncertainty set under Assumption 6.4.

3. Vehicle Dependent Pull Case (VDPC), the intermediate values are simulated from the uncertainty set under Assumption 6.5.

Table 24 shows the results for the rolling horizon used on the previously mentioned scenarios, with their deterministic retrospective counterpart. The results clearly show that the rolling horizon method produces a suboptimal solution. The revenue using the rolling horizon approach may decrease by 20%. Furthermore, a charge violation may occur as can be seen in the IIPC. However, the overall impact on the schedule remains minimal, involving only a 25-minute charging time equivalent charge introduced to the system. Nonetheless, the computational time of rolling horizon with or without sampling is drastically lower compared to the deterministic solutions.

The Figures 33, 35, 37 display the respective schedules that produced the values shown in Table 24. Figures 34 and 36 show the timeline created by uniform Monte Carlo sampling from the uncertainty sets specified in Assumptions 6.4 and 6.5. These are the realized timelines for IIPC and VDPC. The schedule with no Monte Carlo sampling, shown in Figure 33, incorporates fewer grid balancing stints than the Base Case, which is its retrospective counterpart. This shows that the rolling horizon method is inherently suboptimal for the whole timeline. The IICP rolling horizon generated 5 charge violations which summed to be 14.89% of a singular battery capacity. This is equivalent to a total of 25 min of waiting time while the batteries are charging upon the arrival of the 5 vessels.

Table 24: Results of the rolling horizon method, compared to its deterministic counterpart for the different types of uncertainty sets. All optimizations are run with the assumption that the infrastructure consists of 10 batteries and 2 charging stations. The RPO of the no pull case (NPC) corresponds to the Base Case RPO using 10 batteries and 2 charging stations which can be seen in Table 15.

Optimization\ Method	RH	t_{RH} (s)	Charge Violations (%)	RPO	t_{RPO}
NPC	17	733.730	0%	21	2068.750
IIPC	23	1085.450	14.89%	25	3229.031
VDPC	16	1131.810	0%	22*	3600.140

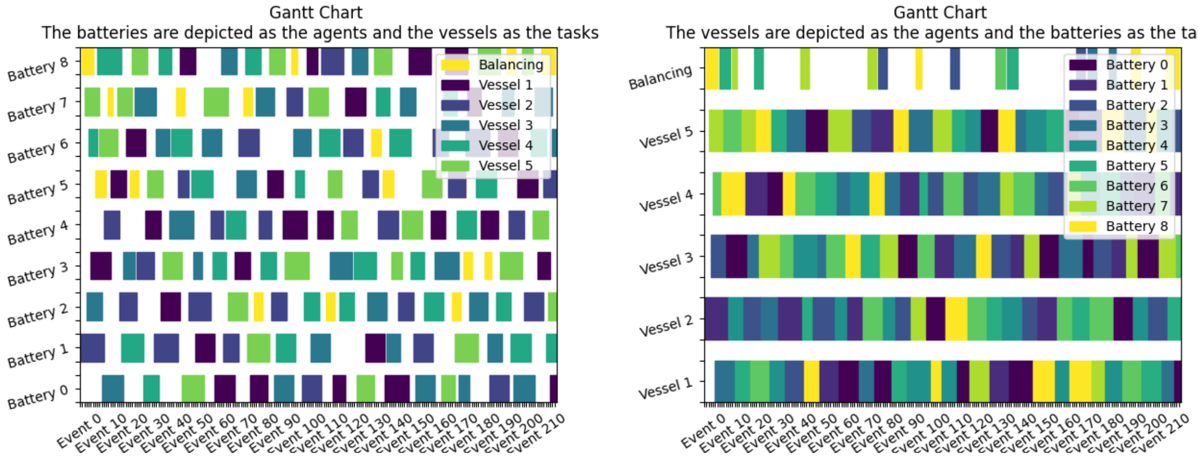


Figure 33: Gantt charts of the Rolling Horizon solution of the Base Case without Monte Carlo sampling. Its retrospective counterpart is equal to Base Case 6.1.1.

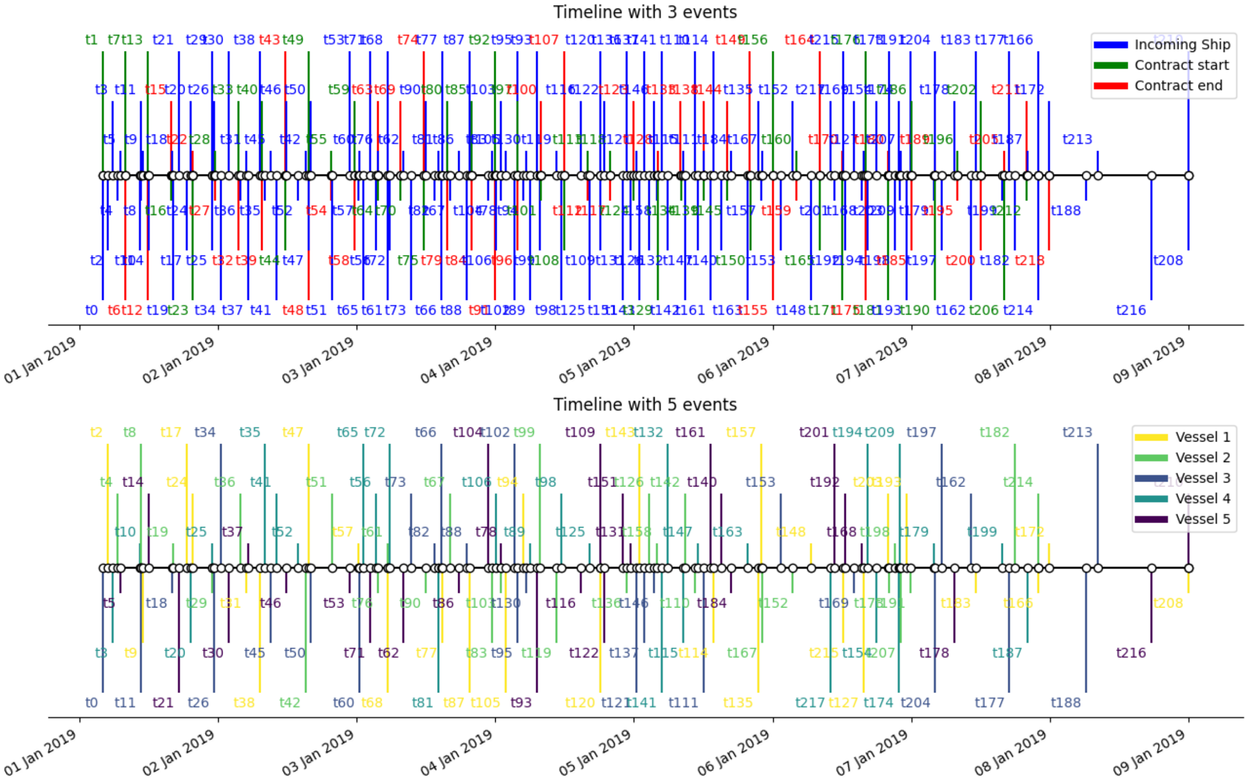


Figure 34: The retrospective timelines from the rolling horizon simulation employing the uncertainty sets as specified in Assumption 6.4. The top figure presents the timeline incorporating the grid balancing stints. The bottom figure illustrates the arrival times of the various vessels.

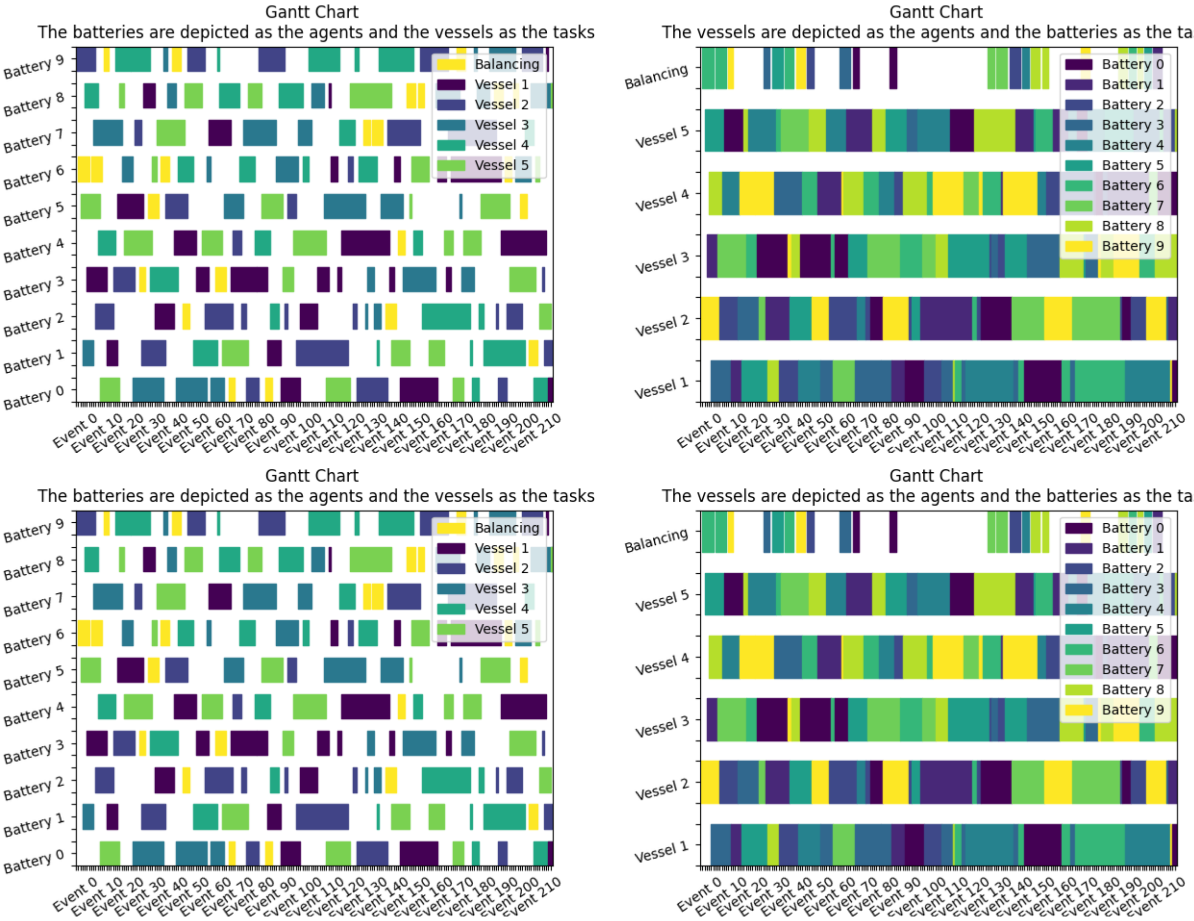


Figure 35: Gantt charts of the rolling horizon simulation of IIPC, which employs the uncertainty sets as specified in Assumption 6.4. The upper figure shows the schedule determined by rolling horizon, the lower figure shows the schedule of its retrospective deterministic version using RPO.

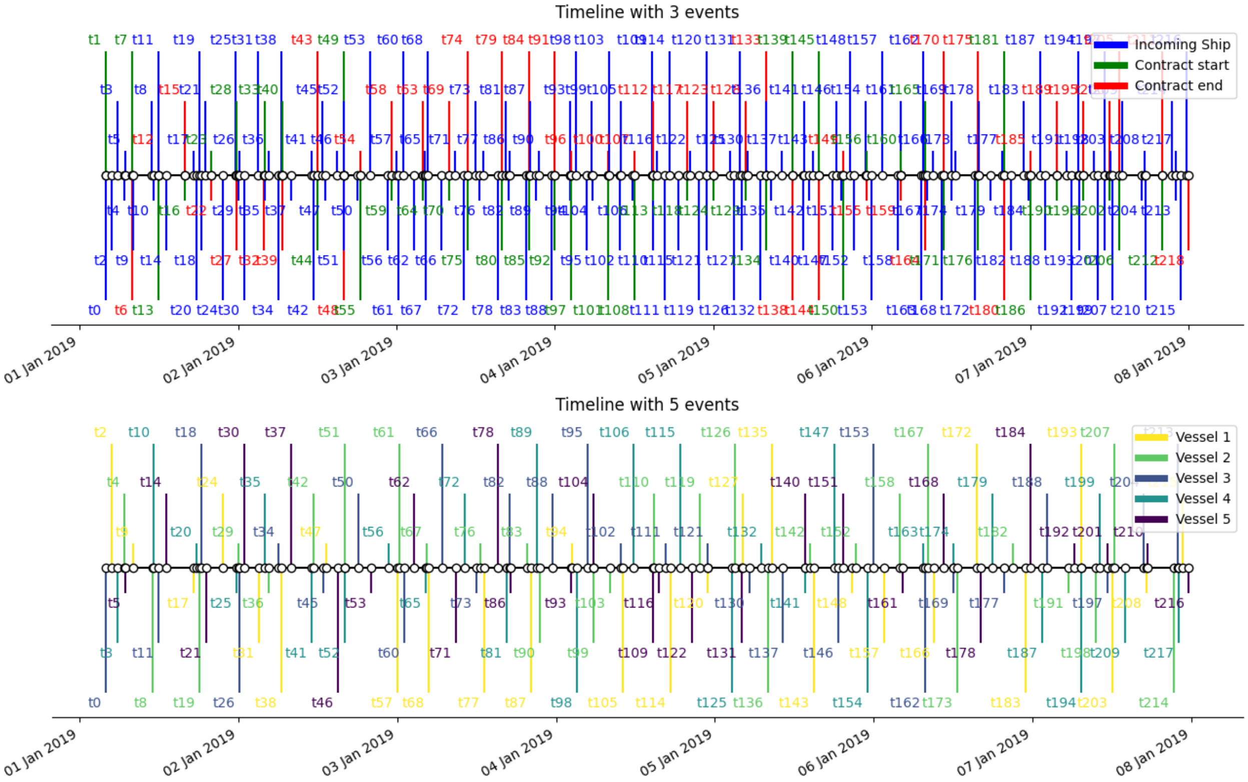


Figure 36: The retrospective timelines from the rolling horizon simulation employing the uncertainty sets as specified in Assumption 6.5. The top figure presents the timeline incorporating the grid balancing stints. The bottom figure illustrates the arrival times of the various vessels.

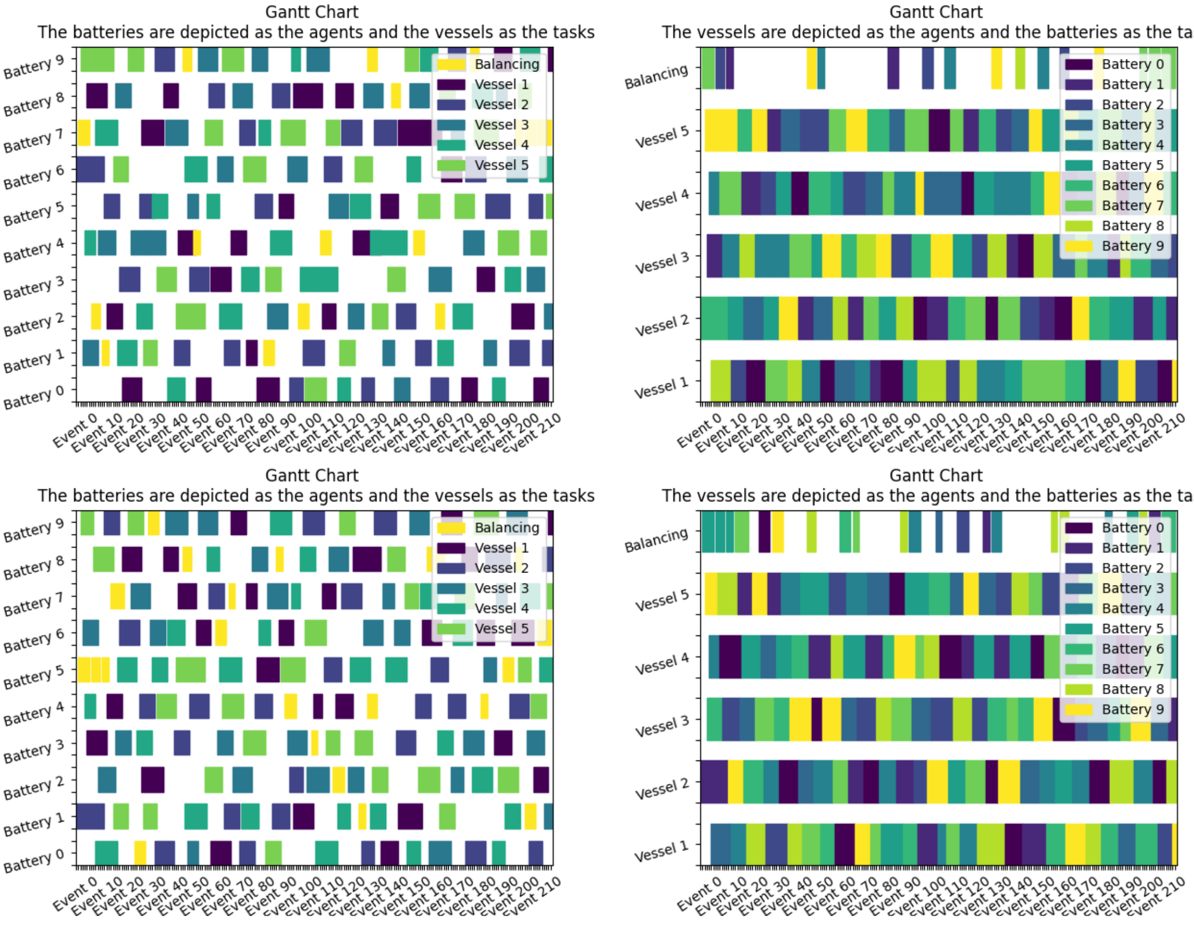


Figure 37: Gantt charts of the rolling horizon simulation of the VDPC which employs the uncertainty sets as specified in Assumption 6.5. The upper figure shows the schedule determined by rolling horizon, the lower figure shows the schedule of its retrospective deterministic version using RPO.

The value at risk time arrival (VAR) analysis only takes a single scenario into consideration. This scenario takes the Base Case and uses Assumption 6.3 to split the timeline according to the method discussed in Section 5.3.2. Figure 38 shows the first day of the timeline created with the value at risk method. Table 25 shows the results of the Base Case compared to the value at risk scenario under Assumption 6.3. Due to the increase in the time dimension, the RPO optimization takes longer to complete and is cut short by the 1-hour time limit. Figures 39 and 40 show two representations of the same schedule that uses 13 grid balancing spots found for VAR. Figure 39 the x-axis is event-based whereas Figure 40 shows the same information with an actual time indication. In Figure 40, it clearly shows that all the batteries are allocated simultaneously to the 5 vessels, this shows that every ship has two allocated batteries within its critical arrival time interval.

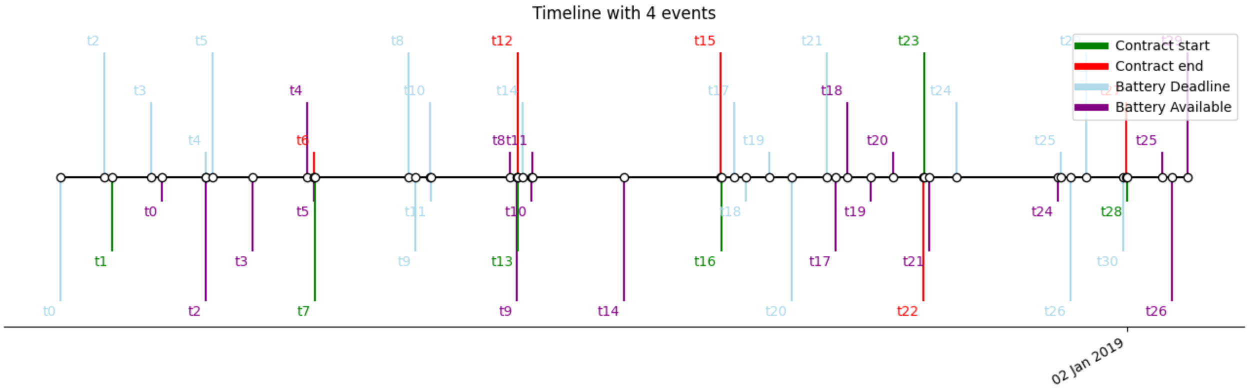


Figure 38: Split timeline for value at risk analysis of the arrival time uncertain case. Only the first day is shown in this figure, the full timeline can be found in Appendix C. Note that the arrival time annotations appear in both light blue and purple.

Table 25: Results of the value at risk analysis compared to Base Case. The results of Base Case as seen in Tables 9 and 10 in Section 6.1.3. Note that both RPOs are performed with 10 batteries and 2 charging stations.

Case \ Methods	BPO	t_{BO} (s)	t_{CSO} (s)	RPO	t_{RPO} (s)	Gap (%)
Base Case	(9, 2)	576.516	67.797	21	2068.750	0%
VAR	(10,2)	45.813	2824.484	13*	3600.047*	146%

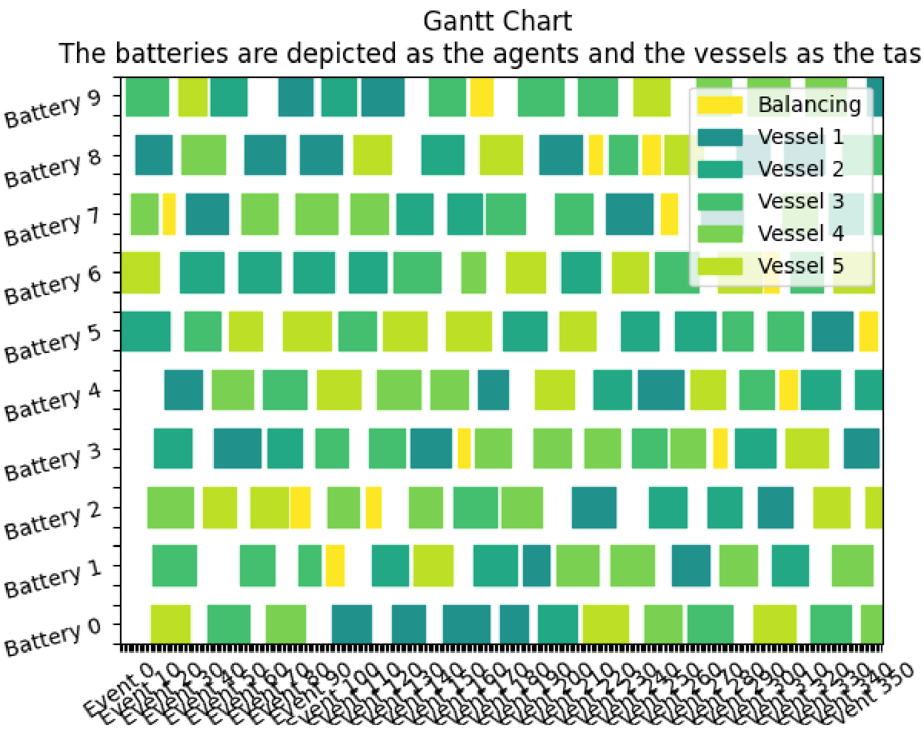


Figure 39: Gantt chart of the value at risk arrival time uncertainty (VAR) scenario where the batteries are depicted as the agents

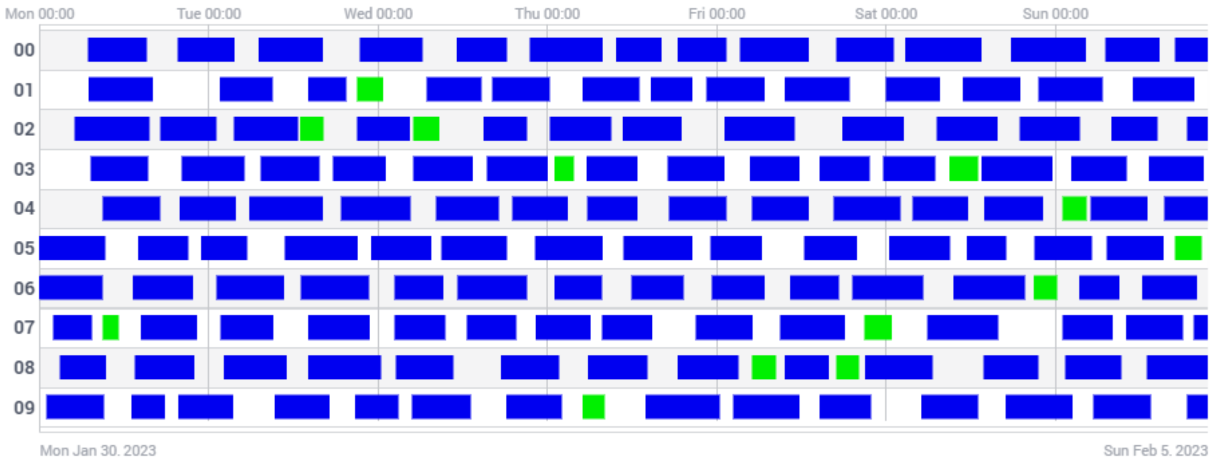


Figure 40: Gantt chart of the value at risk arrival time uncertainty (VAR) scenario where the batteries are depicted as the agents plotted in real-time. The vessels are shown in blue and the grid balancing is shown in green.

7 Conclusion

This section will start with an overview of the problem, the optimization approaches and the obtained results. Subsequently, the main limitations of the assumptions and methods used in this research are discussed. Based on these limitations, potential areas for future research exploration are proposed.

7.1 Overview

In every aspect of transportation, effort should be made to replace fossil fuels with alternative fuels that are renewable and sustainable. Marine transportation is one of the largest polluters and requires a whole system change to facilitate a transition to renewable energies. This research explored the opportunity to transition inland freight vessels to become container battery-powered. The new system utilizes modular exchangeable containers filled with batteries that will be handled at the port for recharging and loading. The battery containers are charged at a charging or docking station in or near the port facilities within the system. While on the charging station, the batteries may be used for alternative revenues such as grid balancing (net stabilization). Two core uncertainties of the maritime sector, namely energy consumption and arrival time uncertainty, are incorporated into the system.

The main objective of research for this thesis was to develop a method that provides port operators with a deeper understanding of the required infrastructure for a new electrified waterway system for freight vessels. This method provides a schedule that optimally uses this infrastructure in this system. A schedule must provide all the vessels in the system with a charged battery upon arrival at the charging port. Arbitrary revenue options for batteries and charging stations are supported by this method. The revenue options are used to additionally maximize external profits. The external profits are formulated as grid balancing positions that require both a battery and a charging station. Grid balancing has the potential to consume or provide electricity to the allocated battery.

The system is modeled as a discrete event time-space, in which the batteries situated at the port are allocated to the task that starts at the specific time event. Constraints on the charge level of the batteries and the capacity of charging stations are formulated. These constraints regulate the amount of energy charged or self-discharged during intervals between two consecutive time events.

This thesis proposes an approximation algorithm, a MIP formulation, and five stochastic programming extensions to the MIP formulation. All the methods try to find the optimal number of batteries, charging stations, and grid balancing revenue of an arbitrary waterway system in different contexts. The approximation algorithm and the MIP formulation find feasible solutions for all counts only using a deterministic case. The direct Monte Carlo simulation extends the MIP formulation to find the counts for all cases that lie within the predefined uncertainty. Rolling horizon adapts the system to step-by-step gather more information through Monte Carlo sampling and solving partial event time-spaces. This method closely relates to real-life information gathering, where more information is added within the run. Probabilistic Constraint assumes the revenue gained by the system and minimizes the probability of failure at any time event. Extreme value analysis assumes the extreme values of the energy consumption uncertainty and creates the boundaries of viable scenarios. Value at risk analysis splits the event time-space in both extremes of the arrival time uncertainty and assumes a scenario satisfying the worst of both outcomes.

The results show that the optimization approaches produce an optimal or near-optimal schedule in their respective contexts, with a single exception of direct Monte Carlo simulation. The key findings of these approaches are listed below.

The approximation algorithm is solved extremely fast but produces unreliable results in its estimates. Given certain cases the solutions produced by the approximation algorithm are overestimated. This approach is very suitable for getting an insight into the upper bound for the optimal number of batteries and charging stations that should be employed in a system. The approximation algorithm is unsuitable for hourly planning due to its limited perspective on the broader context.

The MIP formulation performs well for small experimental data cases and produces schedules that utilize the optimal count of batteries, charging stations, and revenue. However, the method scales very poorly and is computationally heavy, even to the point of exceeding reasonable solving time limits. Performing a sensitivity analysis of the input parameters of the system gives insights into how the optimal values react in different input scenarios. Additionally, the sensitivity analysis identifies areas where the system can easily be improved upon. It shows the influence of critical factors such as the charging speed and a different assortment of revenue options upon the potential of the outcomes.

The computational time encountered by solving the MIP formulation renders the direct Monte Carlo simulation impractical for the cases examined in this research. The number of simulations that can be done in a reasonable time is too few to draw any conclusions from the results found. Therefore, this method was not further explored in the experiments.

The rolling horizon approach is very versatile in incorporating uncertainty and significantly decreases the computational time. The approach emulates a real-life scenario and can easily be implemented in practice. However, the system may find substantially worse answers and suffers from connectivity challenges between different horizon optimizations.

The probabilistic constraint method produces optimal schedules for the revenues in the sense of minimizing the chance of failure. This provides an overview of the possibility of failure occurring at any time event. However, the approach relies on strong assumptions regarding the normal distribution of uncertainty, which is rarely the case in reality.

Extreme value analysis is inherently the same as the MIP formulation approach and therefore suffers from computational heaviness. However, in the context of uncertainty, it does provide boundaries that could limit the search space for other approaches such as the probabilistic constraint approach.

The value at risk approach forms a great tool to check the robustness incorporated within time uncertainty. It provides an indication of the flexibility available within the event time-space.

In summary, the conclusion is that the approximation algorithm is the appropriate first step. The MIP formulation provides a valuable additional step in insight into the complexities of the system. However, it scales too poorly to extend to bigger data cases or to integrate uncertainties. In scenarios involving uncertainty, the rolling horizon approach is recommended due to its adaptability and realistic modeling, with secondary limit analyses of both value at risk and extreme value analyses. The probabilistic constraint approach is a suitable alternative if normally distributed uncertainty is inherent to the uncertainty data.

In general conclusion, the model with these optimization methods provides port operators with essential insights into the system they are managing. An analysis of a real-life situation with these approaches empowers the business case to transition an inland waterway system to become fully container battery-powered.

7.2 Limitations and Further Research

The research inherently introduces three categories of limitations: missing data, model assumptions, and limitations associated with the chosen optimization approaches. This section shows the limitation in the specified order and recommends further research to extend the findings presented in this thesis.

The model is only tested on simulated data, for further research I would recommend running the model on actual arrival times that can be retrieved from historical data. Large databases of historical data exist in the form of AIS-data on inland shipping, such as the data collected by Cofano[7] and MarineTraffic[8]. This data can be refined and the arrival times of vessels at the charging port can be extracted. This information can be used to build a system with real vessels and accurate uncertainty assumptions in both time and energy consumption. The arrival time can be extracted by the timestamps included in the AIS-data. Approximations for energy can be made by extracting the distance traveled and the approximate weight, these can be combined to deduce the energy needed for a voyage. The historical data is left outside of the scope of this research due to the time it takes to refine the data. Accurate implementation eliminates the need for all uncertainty set assumptions (6.2 - 6.5).

The revenues of the inland shipping industry are also left outside the scope of this research due to a lack of data. This can be included lifting the requirement of facilitating all vessels with batteries and treating every task as a revenue option. Alternatively, revenues can be generated by ensuring timely arrivals and penalties for extended waiting times could be incorporated.

In the model description in Section 2.2.1, we made several assumptions about the inland waterway system to simplify the problem. The effects of the relaxation of the most influential assumptions are listed below.

Firstly, Assumption 2.1 assumes that there is just a single charging port. To investigate a system with multiple ports with charging stations, extensions to the model are necessary. Not all batteries that are on land can be allocated to all tasks, therefore both batteries and tasks are location dependent at all time events. This further increases the size of the model. Moreover, this system adds more challenges due to the intricacies of the placement of the batteries and the accumulation of batteries in certain sections of the system. Alternative transport of the container batteries between ports is necessary to avoid over-dimensioning the number of batteries, with its own costs. Also, additional uncertainty is introduced by not knowing the next charging port destination of the vessels included in the system.

Secondly, Assumption 2.10 can be easily implemented by changing the constraint 1 to be equal to the number required at the time event. However, it does ask for an additional assumption on the energy consumption spread over these batteries.

Thirdly, some uncertainties that characterize the maritime sector are excluded by Assumption 2.13. These uncertainties, identified by Ksciuk et al. [2022], are port duration, spot rate revenue, supply, demand, and weather. These uncertainties have an effect on refining the historical data to be able to generate a distribution for energy consumption and arrival times uncertainties. Also, the sampling methods are likely to become dependent on these additional uncertainties.

Fourthly, the harboring times and waiting times at the charging port are excluded from this research

[7] https://www.cofano.nl/nl/

[8] https://www.marinetraffic.com/en/ais/home/

by Assumptions 2.5 and 2.6. These can be implemented into a model by also treating harboring as a task that must be performed by the batteries before they can be attached to a charging station and adding violations to the formulation that represent the waiting times at arrival. Adding these times asks for an increased knowledge of the ship harboring practices. Moreover, another level of uncertainty is added which would require an even larger model, that likely will further increase the computational cost.

Lastly, the equivalence Assumptions 2.8 and 2.9 might be necessary to be relaxed due to net congestion or performance differences. This creates a time and pairwise battery charging station dependence on the charging rate d. Also, charging a battery might not be linear which would turn Constraint 7 into a nonlinear constraint. This would require a nonlinear solver that most likely even further increases the computational time. More data needs to be collected on a specific data case to make these insights worthwhile.

The approaches used in this thesis also carry their limitations. Most of the approaches stem from stochastic programming. Bertsimas and den Hertog [2022] state the following disadvantages on the shortcomings of stochastic programming:

1. the distribution of the uncertain data is often difficult to specify,
2. the probabilities that appear in the problem are often hard to calculate,
3. the resulting feasible regions are often non-convex, and
4. the computer traceability is often problematic because of the large number of scenarios that are used.

The first and fourth comments are not encountered in this research due to the missing data and the inability to optimize a large number of scenarios. The second comment is encountered by the probabilistic constraint approach, where very strict assumptions on the distributions are necessary to be able to calculate the desired probability. The third comment does not apply due to the reliance on integer programming: the feasible regions are never convex.

The approximation algorithm is only tested on the deterministic case with either no or all grid balancing stints included in the input. Multiple extensions of the approximation algorithm can be explored. It could be tightened by starting from the back of the timeline to better evaluate the number of charging stations necessary. By iteratively adding and deleting grid balancing spots, the approximation algorithm can showcase the transition of the objective functions, i.e. the number of batteries and charging stations necessary for a system with some grid balancing spots included. Also, the usage of the approximation algorithm can be explored in cases with uncertainty.

The MIP formulation is highly dependent on the number of batteries due to the decision variables being defined for every battery on every time event. This results in being unable to investigate the Pareto Optimal solutions with a lot of batteries. The limiting factor in optimizing the MIP formulation is the computational time it requires. Only a single set of parameter cuts is explored in this thesis. Surely more system characteristics can be exploited to improve the LP bounds even further. Another method to cut down computational time is feeding the approximation algorithm or any other polynomial approximation solution as a starting point of MIP's branch and bound. This eliminates initial branches that have an LP relaxation higher than the approximated solution. The results of the computational analysis, see Section 6.1.6, clearly show the impact of providing an initial solution on reducing computational time to solve the MIP.

The Monte Carlo sampling methods, direct Monte Carlo simulation and rolling horizon, are only tested with a single scenario. To better cover the uncertainty a lot more simulations should be performed. This also requires the MIP formulation to be faster. The horizon length and save length of rolling horizon can also be optimized on computational time and objective values. Moreover, rolling horizon can also be used to incorporate different decision deadlines, where ship allocations can be decided at the moment of arrival but grid allocations are decided in advance. This extension brings the simulation even closer to real-world scenarios.

The critical values β_b^t are linearized to be able to maintain the linearity of the program, this changes the resulting probabilities. This method is improved by applying quadratic optimization methods.

Extreme scenarios are all logically constructed. Further research can implement an adversary that optimizes the worst cases. The value at risk should be tested with more types of uncertainty sets.

Further research might explore the application of different types of optimization methods suitable for this problem such as heuristics and machine learning algorithms, due to the size of this research these have been omitted.

Given a realistic data characteristic, further research does best to cherry-pick the method extensions that best fit the data. The recommended basic setup is to formulate an approximation method as well as a MIP formulation.

References

Bureau voorlichting binnenvaart. https://bureauvoorlichtingbinnenvaart.nl/de-binnenvaart/basiskennis/vaarwegen/, Accessed: 13-08-2023.

Reducing emissions from the shipping sector. https://climate.ec.europa.eu/eu-action/transport-emissions/reducing-emissions-shipping-sector_en, Accessed: 16-06-2023.

H. Aaltonen, S. Sierla, V. Kyrki, M. Pourakbari-Kasmaei, and V. Vyatkin. Bidding a battery on electricity markets and minimizing battery aging costs: A reinforcement learning approach. *Energies*, 15(14):4960, 2022.

S. Bengtsson, K. Andersson, and E. Fridell. A comparative life cycle assessment of marine fuels: liquefied natural gas and three other fossil fuels. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, 225(2):97–110, 2011. doi: 10.1177/1475090211402136. URL https://doi.org/10.1177/1475090211402136.

D. Bertsimas and D. den Hertog. *Robust and Adaptive Optimization*. Dynamic Ideas LLC, 2022. ISBN 9781733788526.

D. Bertsimas, D. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53, 10 2010. doi: 10.1137/080734510.

C. Coello, D. Veldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems Second Edition*. 01 2007. ISBN 978-0-387-33254-3. doi: 10.1007/978-0-387-36797-2.

S. A. Cook. The complexity of theorem-proving procedures. page 151–158, 1971. doi: 10.1145/800157.805047. URL https://doi.org/10.1145/800157.805047.

G. B. Dantzig and M. N. Thapa. *The simplex method*. Springer, 1997.

G. Erichsen, T. Zimmermann, and A. Kather. Effect of different interval lengths in a rolling horizon milp unit commitment with non-linear control model for a small energy system. *Energies*, 12(6), 2019. ISSN 1996-1073. doi: 10.3390/en12061003. URL https://www.mdpi.com/1996-1073/12/6/1003.

M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA, 1979. ISBN 0716710455.

B. L. Gorissen, İ. Yanıkoğlu, and D. den Hertog. A practical guide to robust optimization. *Omega*, 53:124–137, jun 2015. doi: 10.1016/j.omega.2014.12.006. URL https://doi.org/10.1016%2Fj.omega.2014.12.006.

IEA. Energy technology perspectives 2020, 2020. URL https://www.iea.org/reports/energy-technology-perspectives-2020.

J. Ksciuk, S. Kuhlemann, K. Tierney, and A. Koberstein. Uncertainty in maritime ship routing and scheduling: A literature review. *European Journal of Operational Research*, 2022. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2022.08.006. URL https://www.sciencedirect.com/science/article/pii/S037722172200649X.

L. A. Levin. Universal sequential search problems. *Problems of Information Transmission*, 9(3), 1973.

R. Marler and J. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26:369–395, 04 2004. doi: 10.1007/s00158-003-0368-6.

S. Martello and P. Toth. *Knapsack problems: algorithms and computer implementations.* John Wiley & Sons, Inc., 1990.

N. A. Q. Muzir, M. R. H. Mojumder, M. Hasanuzzaman, and J. Selvaraj. Challenges of electric vehicles and their prospects in malaysia: A comprehensive review. *Sustainability*, 14(14), 2022. ISSN 2071-1050. doi: 10.3390/su14148320. URL https://www.mdpi.com/2071-1050/14/14/8320.

V. Pareto. *Manuale di economia politica.* Piccola biblioteca scientifica. Societa Editrice, 1906. URL https://books.google.nl/books?id=_oJIAAAAYAAJ.

M. Perčić, N. Vladimir, and M. Koričan. Electrification of inland waterway ships considering power system lifetime emissions and costs. *Energies*, 14(21), 2021. ISSN 1996-1073. doi: 10.3390/en14217046. URL https://www.mdpi.com/1996-1073/14/21/7046.

M. Pióro and D. Medhi. Chapter 5 - general optimization methods for network design. In M. Pióro and D. Medhi, editors, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, The Morgan Kaufmann Series in Networking, pages 151–210. Morgan Kaufmann, San Francisco, 2004. doi: https://doi.org/10.1016/B978-012557189-0/50008-1. URL https://www.sciencedirect.com/science/article/pii/B9780125571890500081.

V. V. Vazirani. *Bin Packing*, pages 74–78. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-662-04565-7. doi: 10.1007/978-3-662-04565-7_9. URL https://doi.org/10.1007/978-3-662-04565-7_9.

X. Yan, K. Wang, Y. Yuan, X. Jiang, and R. R. Negenborn. Energy-efficient shipping: An application of big data analysis for optimizing engine speed of inland ships considering multiple environmental factors. *Ocean Engineering*, 169:457–468, 2018. ISSN 0029-8018. doi: https://doi.org/10.1016/j.oceaneng.2018.08.050. URL https://www.sciencedirect.com/science/article/pii/S0029801818316421.

A Appendix Complexity

Definitions

Decision Problems

Definition A.1 (Partition)**.** We say that Q is a *partition* of some set A if:

- Q is a collection of non-empty sets: $Q \subseteq \mathcal{P}(A)$
- No element occurs in two of these subsets: $\forall A_1, A_2 \in Q, A_1 \cup A_2 = \emptyset$ or $A_1 = A_2$
- Every element occurs once: $\bigcap Q = A$

Definition A.2 (Decision problem)**.** A *decision problem* Π is a yes or no problem that is given by a set of instances $\mathcal{I}$. An instance of the problem specifies the following:

- a set of *feasible* solutions $\mathcal{F}$ for I;
- a cost function $c : \mathcal{F} \to \mathbb{R}$;
- a limit constant k.

The objective of the problem is to determine if the exists a $F \in \mathcal{F}$ such that $c(F) \leq k$. If the F exists for instance $\mathcal{I}$ we have a *yes-instance*, otherwise, we call I a *no-instance*.

Definition A.3 (Boolean satisfiability problem)**.** he *Boolean satisfiability problem* is the yes or no problem (decision problem) given by a function F consisting of n variables $x_1, \cdots, x_n$ and Boolean operators. The objective is to find assignments to let the expression both be true and false. If both assignments exist, the instance is called a yes-instance otherwise it is called a no-instance.

Definition A.4 (Partition problem)**.** The *Partition problem* is the yes or no problem (decision problem). The objective is to partition the set $S = \{s_1, \cdots, s_n\}$ into subsets S_1, S_2 such that $\sum i \in S_1 s_i = \sum j \in S_2 s_j$. If such a partition exists, the instance is called a yes-instance otherwise it is called a no-instance.

Definition A.5 (3-Partition problem)**.** The *3-Partition problem* is the yes or no problem (decision problem) given by a set A of $3k$ elements and mapping $w : A \to \mathbb{N}$ of weights and $\beta \in \mathbb{N}$. The question is if there exists a partition $\{S_1, \cdots, S_m\}$ (A.1) such that $\sum_{a \in S_1} w(a) = \beta.$, where all sets S_i contain exactly 3 elements. If such a partition exists, the instance is called a yes-instance otherwise it is called a no-instance.

Definition A.6 (Bin Packing problem)**.** Let I be a set of items with size $s : I \to Q \cap (0, 1]$ and let there be a large number of bins with capacity 1. Then, the *bin packing problem* is an optimization problem where the objective is to find the smallest number of bins $k \in \mathbb{N}$ such that there exists a partition of disjoint sets $I_1, \cdots, I_k$ such that the sum of sizes of each item in bin j (i in I_j) is smaller equal to one.

Complexity Classes

Definition A.7 (The complexity class $\mathcal{P}$)**.** The complexity class $\mathcal{P}$ consists of all decision problems for which exists an algorithm that in for every instance I can determine whether it is a yes- or no-instance in polynomial time. The time the algorithm takes to converge is polynomial regarding the input size.

Definition A.8 (The complexity class $\mathcal{NP}$)**.** The complexity class $\mathcal{NP}$ consists of all decision problems with the property that every yes-instance can be verified to be a yes-instance in polynomial time.

Definition A.9 (Polynomial-time reduction)**.** Let $\Pi_1 : \{0,1\}^p \rightarrow \{0,1\}, \Pi_2 : \{0,1\}^q \rightarrow \{0,1\}$ be two decision problems. We say Π_2 *reduces* from Π_1 if there exists a function $\phi : \{0,1\}^p \rightarrow \{0,1\}^q$ that maps every instance of $I_1 \in \{0,1\}^p$ to $I_2 = \phi(I_1) \in \{0,1\}^q$ such that

- the reduction that is polynomially bounded by the input size of I_2.
- I_1 is a yes-instance iff I_2 is a yes-instance.

Informally the polynomial-time reduction from Π_1 to Π_2 implies that Π_1 is 'no harder' than Π_2, in the sense that a polynomial-time algorithm for Π_2 implies a polynomial-time algorithm for Π_1. This is formalized in the Cook-Levin Theorem which can be found at A.16.

Definition A.10 (The complexity class $\mathcal{NP}$-hard)**.** The complexity class of $\mathcal{NP}$-hard consist of all problems Π for which holds that, for every problem $\hat{\Pi}$ in $\mathcal{NP}$ there exists a polynomial-time reduction from $\hat{\Pi}$ to Π.

Definition A.11 (The complexity class $\mathcal{NP}$-complete)**.** The complexity class of $\mathcal{NP}$-complete consist of all problems Π for which holds that:

- $\Pi \in \mathcal{NP}$
- Π is $\mathcal{NP}$-Hard

The class of $\mathcal{NP}$-complete will be split into two subclasses: weakly and strongly NP-complete.

Definition A.12 (Weakly $\mathcal{NP}$-complete)**.** A decision problem is called weakly $\mathcal{NP}$-complete if the problem can be solved in polynomial time (in $\mathcal{P}$) when all the input numbers parameters are represented in unary (Natural numbers).

Definition A.13 (Strongly $\mathcal{NP}$-complete)**.** A decision problem is called strongly $\mathcal{NP}$-complete if the problem remains in $\mathcal{NP}$ even if all input numbers are represented in unary.

Algorithms

Algorithm A.14 (First-Fit for the Bin Packing Problem)**.** *This algorithm looks at the bins one by one and fits the current item in the first bin where there is enough space. The algorithm has the following pseudo-code:*

```
1. For $i$ in Items do
    (a) For $b$ in Bins do
        i. If $i$ fits in $b$: pack $i$ in $b$ and break;
    (b) If $i$ is not packed: generate a new bin $b$ and pack $i$ in $b$.
2. Set $B$ to be the number of bins.
```

Algorithm A.15 (Best-Fit for the Bin Packing Problem)**.** *This algorithm looks at the bins one by one and fits in the bin where there is the tightest fit to the maximum capacity if the current item is added. The algorithm has the following pseudo-code:*

```
1. For $i$ in Items do
    (a) For $b$ in Bins do
        i. Calculate remaining space if $i$ is packed in $b$.
    (b) Pack $i$ in bin $b$ with the least remaining space. If no $b$ exists with enough space: generate
        a new bin $b$ and pack $i$ in $b$.
2. Set $B$ to be the number of bins.
```

Theorems

Theorem A.16 (Cook-Levin Theorem (Levin [1973], Cook [1971]))**.** *The Boolean Satisfiability problem is NP-Complete. Also, all the decision problems that are in NP and polynomially reducible to the Boolean Satisfiability problem are NP-complete.*

Proof. The proof can be found in Garey and Johnson [1979]. □

Theorem A.17 (Law of Large Numbers)**.** *The average result of a large number of trials should be close to the expected value and tends to get closer to the expected value the more trials are performed. The formal mathematical formulation is as follows:*

Let $X_1, X_2, X_3, \cdots$ be a sequence of independent and identically distributed random variables with the distribution function F_x and expected value μ. Denote the sample mean of the first n random variables as $S_n = (X_1 + X_2 + \cdots + X_n)/n$. Then, the theorem states that:

$$S_n \to \mu, \text{ as } n \to \infty$$

B Appendix Model and MIP Formulations

Availability Matrix

Definition B.1 (The availability matrix)**.** The availability matrix A shows the relation of battery availability at consecutive time events. If a battery is assigned to a vessel at time t_1, the same battery is unavailable for assignment for the following k time events. The number k could be different for each t_1. The availability matrix is an upper triangular matrix with a specific number of connected ones on each row. Therefore the matrix has the following form:

$$A_{t_1,t_2} = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 1 & 0 & \cdots & 0 \\ & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ & & \ddots & & \ddots & & & & \\ & & & \ddots & & \ddots & & & \\ & & & & \ddots & & \ddots & & \\ & & & & & \ddots & & \ddots & \\ & & \varnothing & & & & 1 & 1 & 1 \\ & & & & & & & 1 & 1 \\ & & & & & & & & 1 \end{bmatrix}. \tag{41}$$

Definition B.2 (The availability matrix for value at risk time uncertainty)**.** The availability matrix $\hat{A}$ shows the relation of battery availability at consecutive time events in the value at risk arrival time uncertainty case. This matrix is larger than the matrix defined in Equation 41 due to the split of t_a into t_a^- and t_a^+. The rows that correspond with time event t_1^- are similar to the rows seen in the standard A_{t_1,t_2} matrix, the difference lies in that the ones extend to the column corresponding to time event t_2^+. The rows that correspond with time event t_1^+ are zero rows, because no battery is allocated at this event. Therefore the matrix has the following form:

$$\hat{A}_{t_1^-,t_2^+} = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 1 & 0 & \cdots & 0 & \\ & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & \\ & & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \\ & & & \ddots & & \ddots & & & & \\ & & & & \ddots & & \ddots & & & \\ & & & & & \ddots & & \ddots & & \\ & & & & & & \ddots & & \ddots & \\ & & \varnothing & & & & & 1 & 1 & 1 \\ & & & & & & & & 1 & 1 \\ & & & & & & & & & 0 \end{bmatrix}. \tag{42}$$

No Parameter Cuts

Definition B.3 (Feasibility Problem)**.** Given a timeline of arriving ships, we have the following optimization problem:

$$
\begin{aligned}
\text{minimize:}\quad & \left[\sum_b y_b, z\right] & \\
\text{subject to:}\quad & \sum_b x_b^{t_a} &= 1 \\
& \sum_t x_b^t A_{t_1,t} &\leq 1 \\
& c_b^t &\geq C_t \cdot x_b^t \\
& c_b^{t+1} &\leq c_b^t - E_t x_b^t + (1 - x_b^t) \\
& c_b^{t+1} &\leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) \\
& x_b^{t_1} \sum_t (A_{t_1,t}) &\leq \sum_t ((1 - w_b^t) A_{t_1,t}) \\
& c_b^{t_{\min}} &= c_b^0 \\
& x_b^t w_b^t &= 0, \forall t \in t_{\min} \leq t < t_{S_b}, \\
& N y_b &\geq \sum_t x_b^t \\
& M &\geq \sum_b (w_b^t) \\
& y_b^t, x_b^t &\in \{0, 1\} \\
& c_b^t, w_b^t &\in [0, 1]
\end{aligned}
$$

Definition B.4 (Revenue Problem)**.** Given a y_b, M, we have the following optimization problem:

$$\begin{aligned}
\text{maximize: } & \sum_{t_g} \sum_b R_{t_g} x_b^{t_g} \\
\text{subject to: } & \sum_b x_b^{t_a} && = 1 \\
& \sum_t x_b^t A_{t_1,t} && \leq 1 \\
& c_b^t && \geq C_t \cdot x_b^t \\
& c_b^{t+1} && \leq c_b^t - E_t x_b^t + (1 - x_b^t) \\
& c_b^{t+1} && \leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) \\
& x_b^{t_1} \sum_t (A_{t_1,t}) && \leq \sum_t ((1 - w_b^t) A_{t_1,t}) \\
& c_b^{t_{\min}} && = c_b^0 \\
& x_b^t w_b^t && = 0, \forall t \in t_{\min} \leq t < t_{S_b}, \\
& M && \geq \sum_b (w_b^t) \\
& x_b^t && \in \{0, 1\} \\
& c_b^t, w_b^t && \in [0, 1]
\end{aligned}$$

Parameter Cuts

Definition B.5 (Feasability Problem with parameter cuts)**.** We have the following optimization problem:

$$
\begin{aligned}
\text{minimize:} \quad & \left[\sum_b y_b, z\right] && (43)\\
\text{subject to:} \quad & \sum_b x_b^{t_a} &&= 1 && (44)\\
& \sum_b x_b^{t_g} &&\leq 1 && (45)\\
& \sum_t x_b^t A_{t_1,t} &&\leq 1 && (46)\\
& c_b^t &&\geq C_t^- \cdot x_b^t && (47)\\
& c_b^t &&\leq C_t^+ \cdot x_b^t + (1 - x_b^t) && (48)\\
& c_b^{t+1} &&\leq c_b^t - E_t x_b^t + (1 - x_b^t) && (49)\\
& c_b^{t+1} &&\leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) && (50)\\
& x_b^{t_1} \sum_t (A_{t_1,t}) &&\leq \sum_t ((1 - w_b^t) A_{t_1,t}) && (51)\\
& c_b^{t_{\min}} &&= c_b^0 && (52)\\
& x_b^t w_b^t &&= 0, \forall t \in t_{\min} \leq t < t_{S_b}, && (53)\\
& M &&\geq \sum_b (w_b^t) && (54)\\
& U_b^t + I_b^t &&\leq y(b) && (55)\\
& U_b^t &&= \sum_{\hat{t}} x_b^{\hat{t}} \cdot A(\hat{t}, t) && (56)\\
& w_b^t &&\leq I_b^t && (57)\\
& x_b^t &&\in \{0, 1\} && (58)\\
& c_b^t, w_b^t, U_b^t, I_b^t &&\in [0, 1] && (59)
\end{aligned}
$$

Definition B.6 (Revenue Problem with parameter cuts)**.** Given a y_b and M, we have the following optimization problem:

$$\begin{aligned}
\text{maximize:}\quad & \sum_{t_g}\sum_{b} R_{t_g} x_b^{t_g} && (60)\\
\text{subject to:}\quad & \sum_{b} x_b^{t_a} && =1 && (61)\\
& \sum_{t} x_b^t A_{t_1,t} && \leq 1 && (62)\\
& c_b^t && \geq C_t^- \cdot x_b^t && (63)\\
& c_b^t && \leq C_t^+ \cdot x_b^t + (1 - x_b^t) && (64)\\
& c_b^{t+1} && \leq c_b^t - E_t x_b^t + (1 - x_b^t) && (65)\\
& c_b^{t+1} && \leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) && (66)\\
& x_b^{t_1} \sum_{t} (A_{t_1,t}) && \leq \sum_{t} ((1 - w_b^t) A_{t_1,t}) && (67)\\
& c_b^{t_{\min}} && = c_b^0 && (68)\\
& x_b^t w_b^t && = 0, \forall t \in t_{\min} \leq t < t_{S_b}, && (69)\\
& M && \geq \sum_{b} (w_b^t + \sum_{t_g} x_b^{t_g} A_{t_g,t}) && (70)\\
& U_b^t + I_b^t && \leq y(b) && (71)\\
& U_b^t && = \sum_{\hat{t}} x_b^{\hat{t}} \cdot A(\hat{t}, t) && (72)\\
& w_b^t && \leq I_b^t && (73)\\
& x_b^t && \in \{0, 1\} && (74)\\
& c_b^t, w_b^t, U_b^t, I_b^t && \in [0, 1] && (75)
\end{aligned}$$

Rolling Horizon: Parameter Cuts with Charging Violation

Definition B.7 (Revenue Problem with parameter cuts and a charging violation used for rolling horizon analysis)**.** Given a y_b and M, we have the following optimization problem:

$$\text{maximize: } \sum_{t_g}\sum_{b} R_{t_g} x_b^{t_g} - N \sum_{t}\sum_{b} V_b^t \tag{76}$$

$$\text{subject to: } \sum_{b} x_b^{t_a} = 1 \tag{77}$$

$$\sum_{t} x_b^t A_{t_1,t} \le 1 \tag{78}$$

$$c_b^t \ge C_t^- \cdot x_b^t \tag{79}$$

$$c_b^t \le C_t^+ \cdot x_b^t + (1 - x_b^t) \tag{80}$$

$$c_b^{t+1} \le c_b^t - E_t x_b^t + (1 - x_b^t) \tag{81}$$

$$c_b^{t+1} \le c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) + V_b^t \tag{82}$$

$$x_b^{t_1} \sum_{t} (A_{t_1,t}) \le \sum_{t} ((1 - w_b^t) A_{t_1,t}) \tag{83}$$

$$c_b^{t_{\min}} = c_b^0 + V_b^0 \tag{84}$$

$$x_b^t w_b^t = 0, \forall t \in t_{\min} \le t < t_{S_b}, \tag{85}$$

$$M \ge \sum_{b} (w_b^t + \sum_{t_g} x_b^{t_g} A_{t_g,t}) \tag{86}$$

$$U_b^t + I_b^t \le y(b) \tag{87}$$

$$U_b^t = \sum_{\hat{t}} x_b^{\hat{t}} \cdot A(\hat{t}, t) \tag{88}$$

$$w_b^t \le I_b^t \tag{89}$$

$$x_b^t \in \{0, 1\} \tag{90}$$

$$c_b^t, w_b^t, U_b^t, I_b^t \in [0, 1] \tag{91}$$

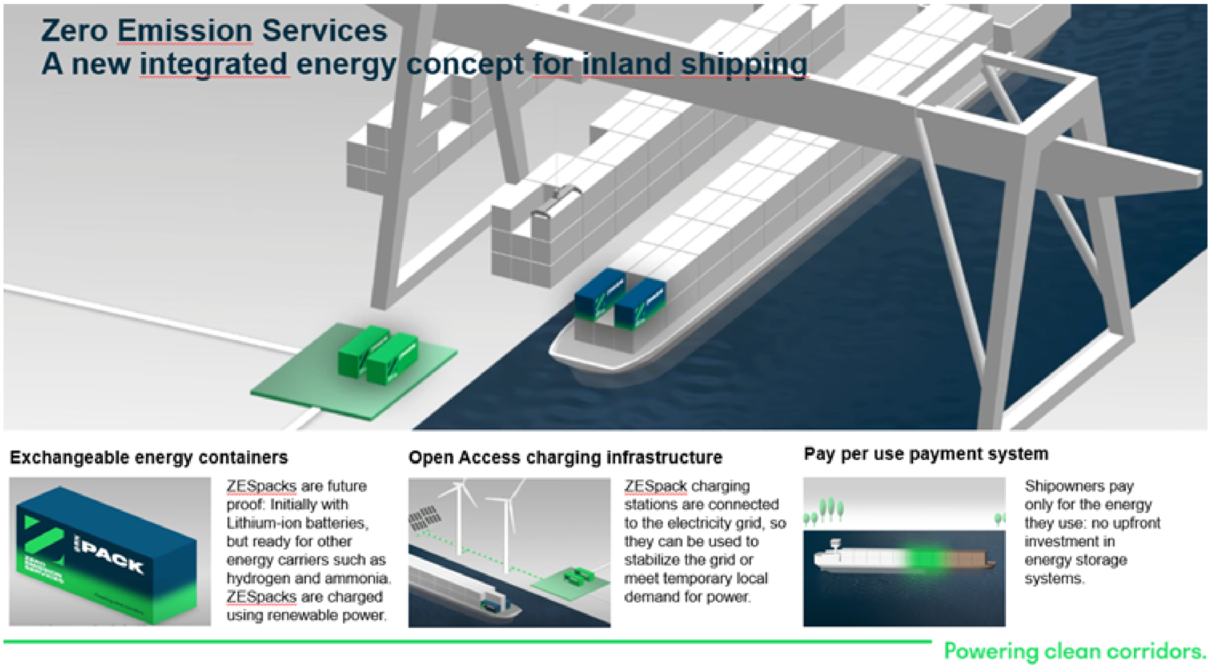
Probabilistic Constraint

Definition B.8 (Revenue Problem with probabilistic constraint)**.** Given a y_b, M, R, $\bar{X}_a^t$, and $\bar{X}_g^t$ we have the following optimization problem:

$$
\begin{array}{rll}
\text{maximize:} & \beta & \\
\text{subject to:} & \sum\limits_b x_b^{t_a} & =1 \\
& \sum\limits_t x_b^t A_{t_1,t} & \leq 1 \\
& c_b^t & \geq C_t \cdot x_b^t \\
& c_b^{t+1} & \leq c_b^t - E_t x_b^t + (1 - x_b^t) \\
& c_b^{t+1} & \leq c_b^t + d\Delta^{t+1} w_b^{t+1} - e\Delta^{t+1}(1 - w_b^{t+1}) \\
& x_b^{t_1} \sum\limits_t (A_{t_1,t}) & \leq \sum\limits_t ((1 - w_b^t) A_{t_1,t}) \\
& c_b^{t_{\min}} & = c_b^0 \\
& x_b^t w_b^t & = 0, \forall t \in t_{\min} \leq t < t_{S_b}, \\
& M & \geq \sum\limits_b (w_b^t) \\
& R & = \sum\limits_{t_g} \sum\limits_b R_{t_g} x_b^{t_g} \\
& \beta & \leq \dfrac{1 + \sum_{t_1 \leq t} \Delta_{t_1} \cdot w_{b_1}^{t_1} - \sum_{t_a \leq t} x_b^{t_a} \mu_{t_a} + \sum_{t_g \leq t} x_b^{t_g} \mu_{t_g}}{\sum_{t_a \leq t} \bar{X}_a^t \sigma_{t_a} + \sum_{t_g \leq t} \bar{X}_g^t \sigma_{t_g}} \\
& x_b^t & \in \{0, 1\} \\
& c_b^t, w_b^t & \in [0, 1]
\end{array}
$$

C Appendix Figures

Industry Information



Zero Emission Services
A new integrated energy concept for inland shipping

Exchangeable energy containers

ZESpacks are future proof: Initially with Lithium-ion batteries, but ready for other energy carriers such as hydrogen and ammonia. ZESpacks are charged using renewable power.

Open Access charging infrastructure

ZESpack charging stations are connected to the electricity grid, so they can be used to stabilize the grid or meet temporary local demand for power.

Pay per use payment system

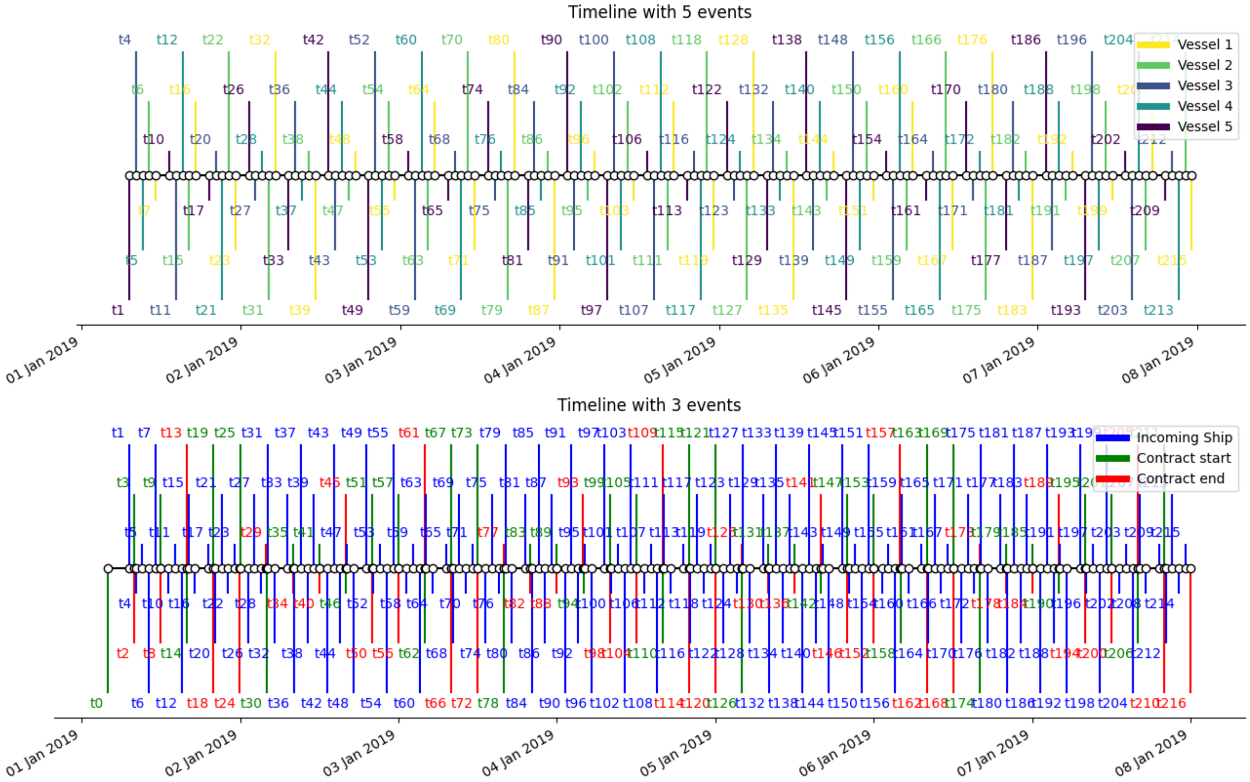
Shipowners pay only for the energy they use: no upfront investment in energy storage systems.

Powering clean corridors.

Figure 41: This figure shows the concept of Zero Emission Services, a commercial supplier of battery containers with the battery swapping service Incorporated. `https://zeroemissionservices.nl/en/homepage/`, Accessed on 16-06-2023

Figures Cyclic Case

Figure 42: Timelines Cyclic Case



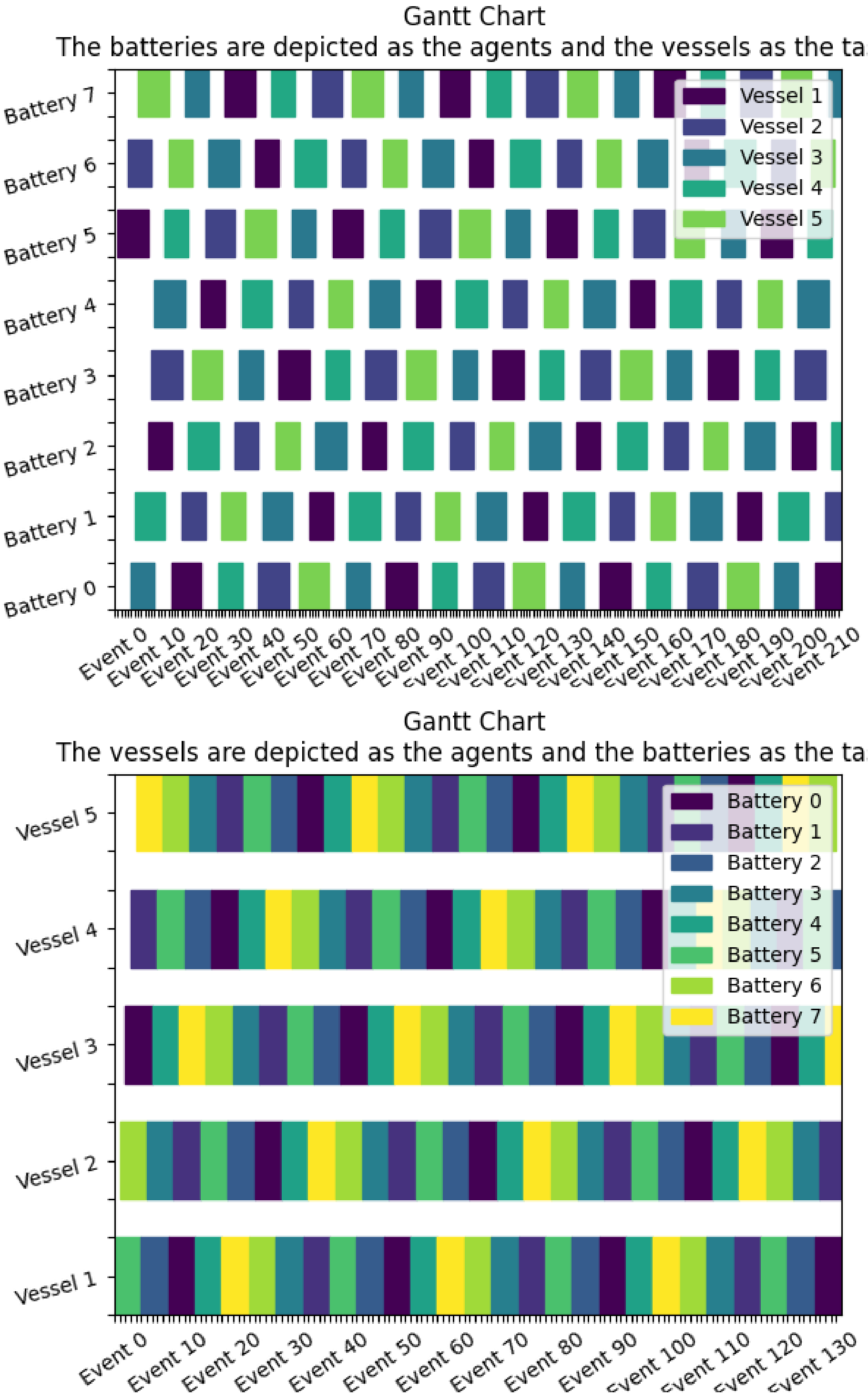


Figure 43: Gantt charts of the solution of the Cyclic Case using the Approximation Algorithm.

Figures Tight Case

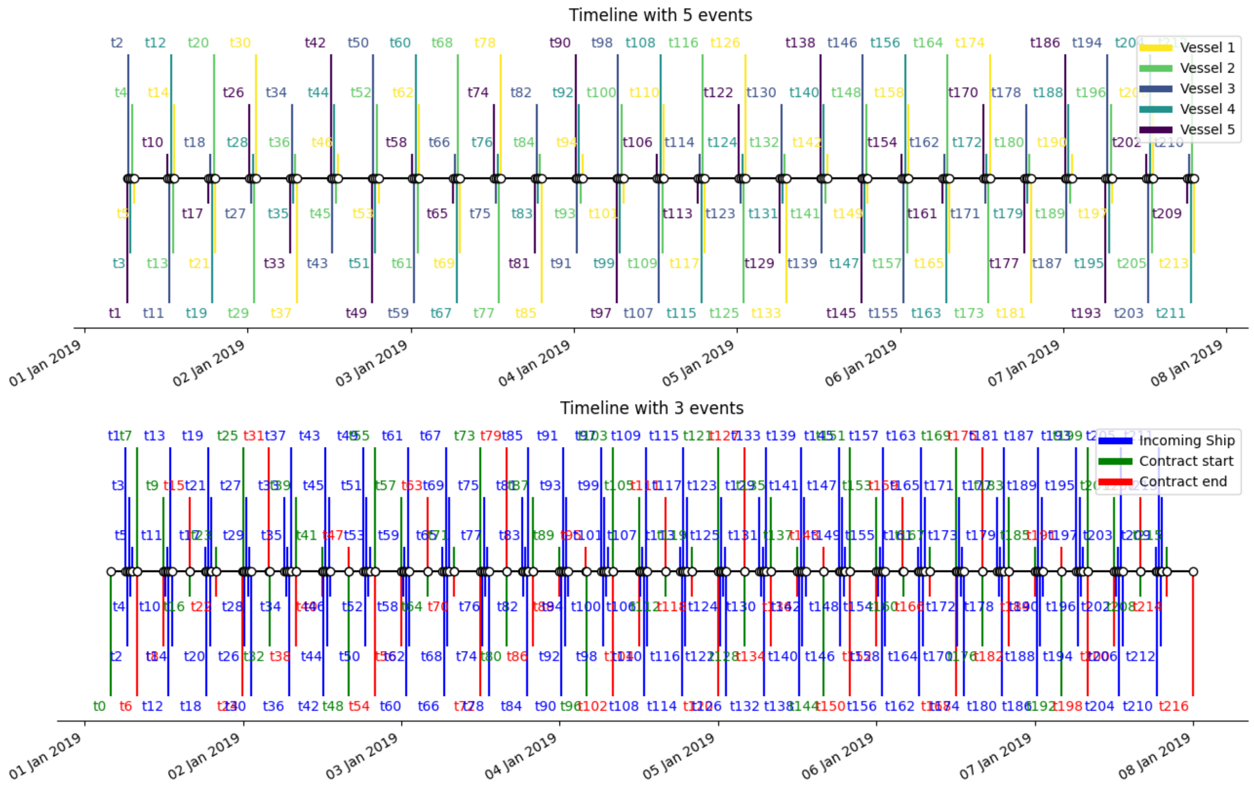


Figure 44: Timelines Tight Case

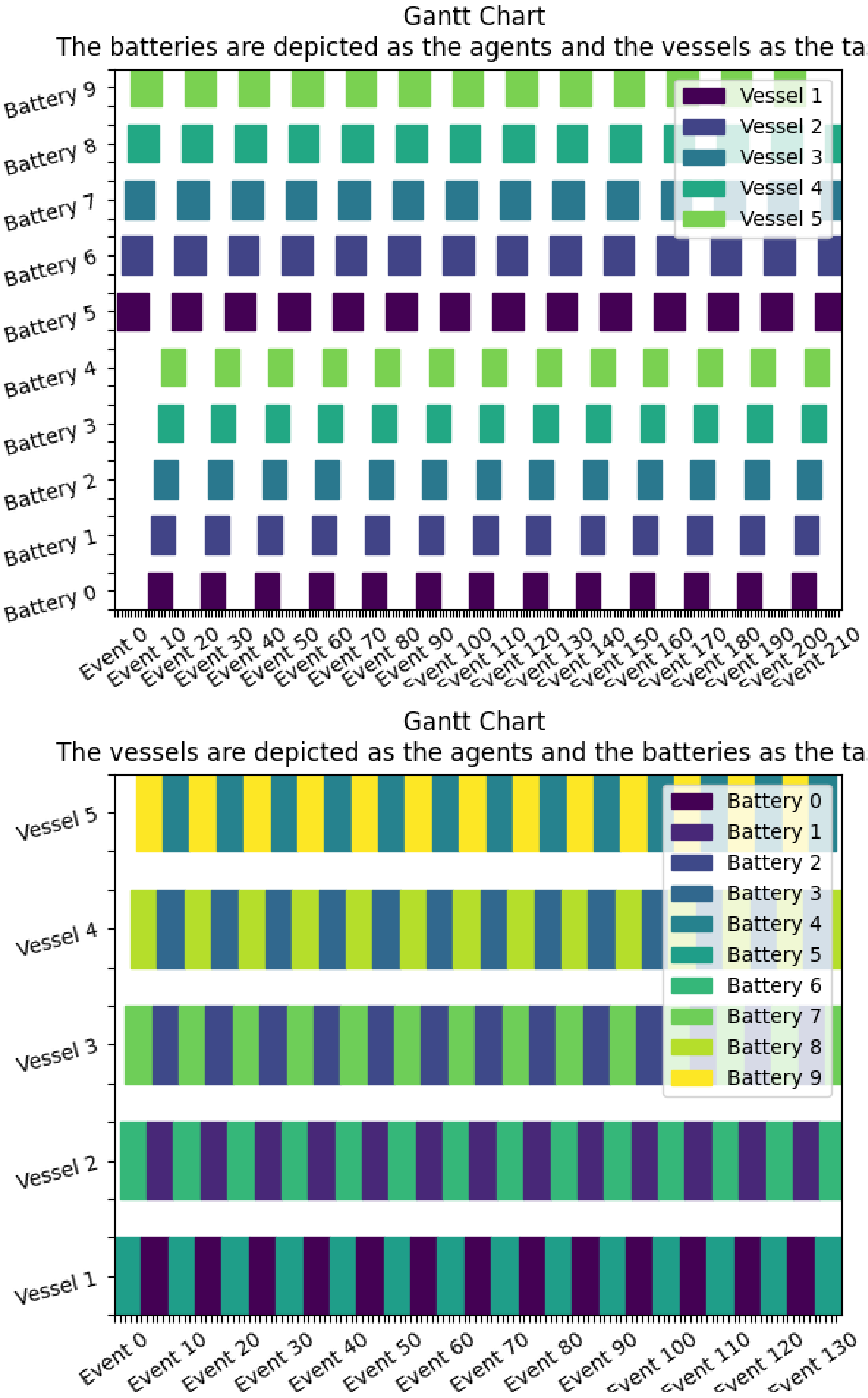


Figure 45: Gantt charts of the solution of the Tight Case using the Approximation Algorithm.

Figures Random Grid Balancing Case

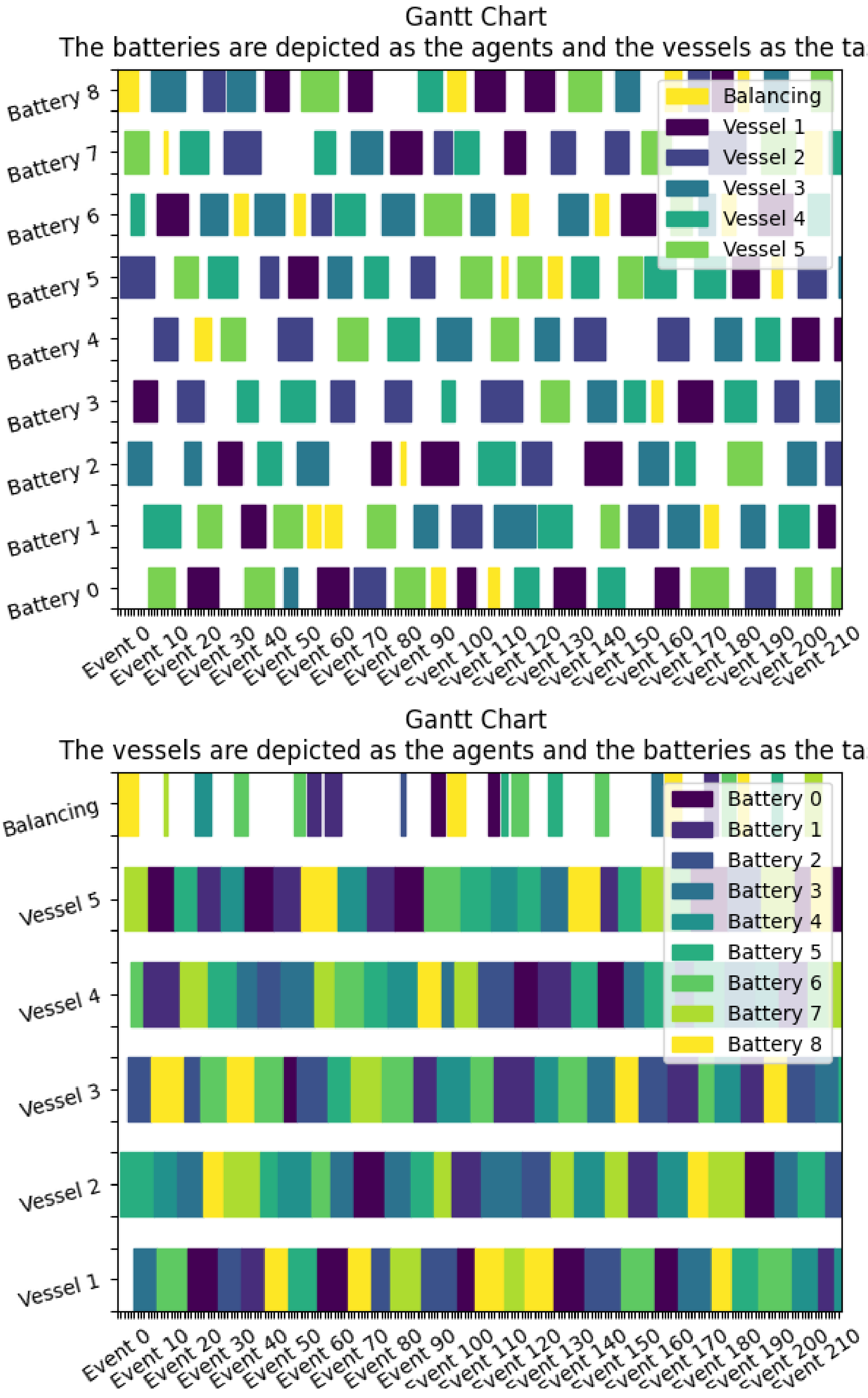


Figure 46: Gantt charts of the random duration of grid balancing stints by RPO.

Figures Time Horizons

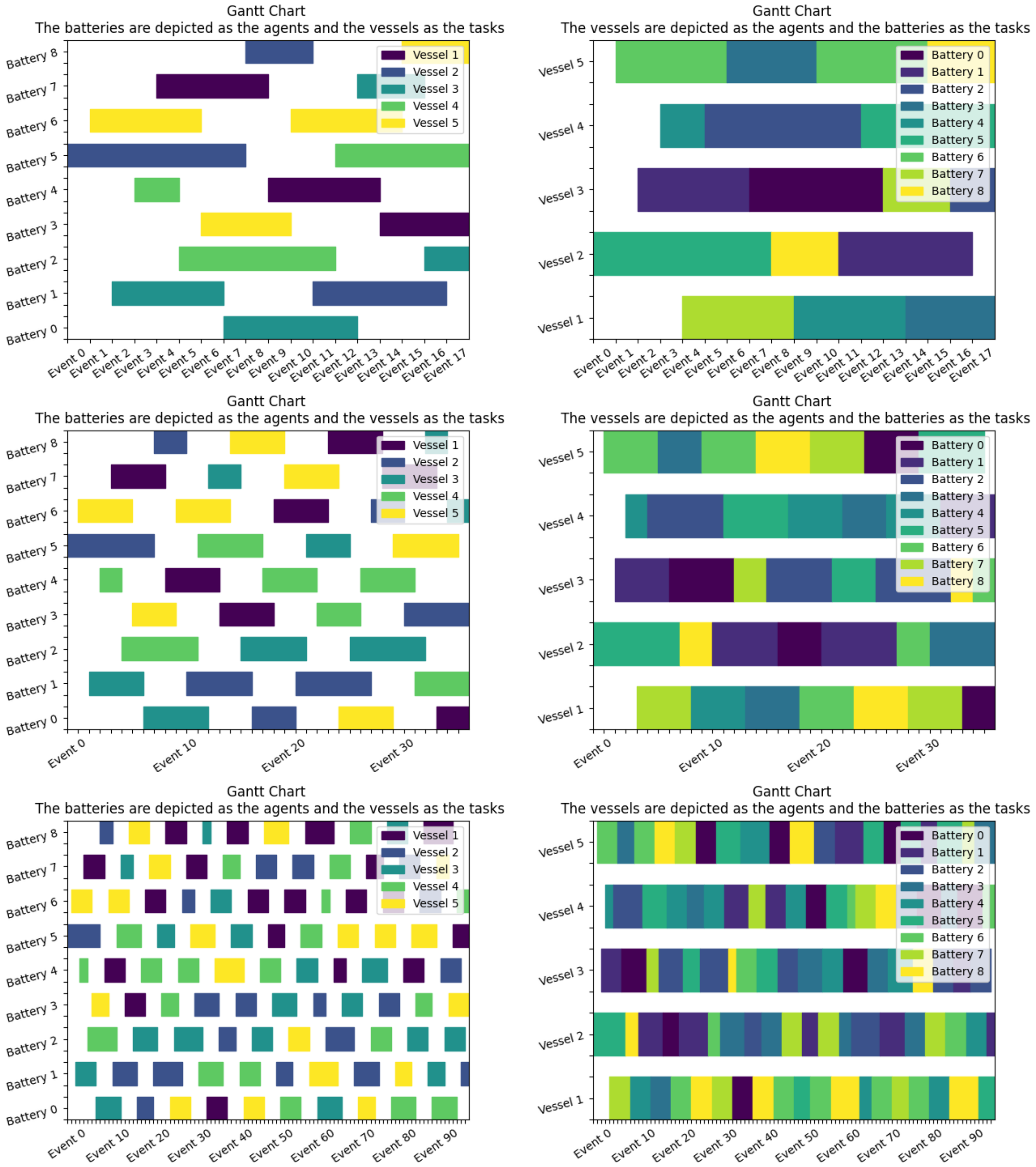


Figure 47: Gantt charts of the different time horizons produced by the approximation algorithm. The different horizons are shown from small to big: the first row of charts has a horizon of one day, the second row of two days, and the final row of five days.

Figures Value At Risk Time Arrival Uncertainty

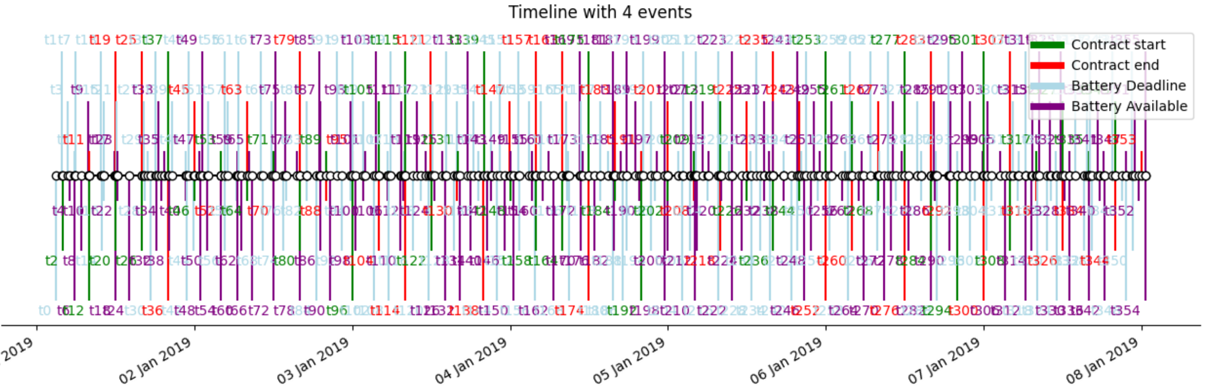


Figure 48: Full timeline of value at risk time arrival uncertainty case (WCTA)

D Appendix Simulation Tables

Table 26: Base Case: Vessel Time Simulation

Time	Ship	Annotation
01/01/2019, 03:58:42	2	t0
01/01/2019, 04:50:27	5	t2
01/01/2019, 05:45:33	3	t3
01/01/2019, 06:50:18	4	t4
01/01/2019, 06:58:50	1	t5
01/01/2019, 10:50:39	4	t8
01/01/2019, 10:58:55	5	t9
01/01/2019, 11:15:27	3	t10
01/01/2019, 11:17:12	2	t11
01/01/2019, 13:05:21	1	t14
01/01/2019, 17:15:18	5	t17
01/01/2019, 17:28:54	2	t18
01/01/2019, 17:57:06	4	t19
01/01/2019, 18:23:11	3	t20
01/01/2019, 19:05:37	1	t21
01/01/2019, 21:38:11	5	t24
01/01/2019, 23:41:19	3	t25
01/01/2019, 23:52:55	2	t26
01/02/2019, 00:11:47	4	t29
01/02/2019, 00:55:08	1	t30
01/02/2019, 03:03:48	5	t31
01/02/2019, 05:32:40	2	t34
01/02/2019, 05:36:27	3	t35
01/02/2019, 06:08:04	4	t36
01/02/2019, 07:21:07	1	t37
01/02/2019, 07:44:34	5	t38
01/02/2019, 11:04:41	3	t41
01/02/2019, 11:22:50	4	t42
01/02/2019, 12:48:03	2	t45
01/02/2019, 14:33:18	1	t46
01/02/2019, 14:58:44	5	t47
01/02/2019, 18:04:08	2	t50
01/02/2019, 18:05:33	4	t51
01/02/2019, 18:07:28	3	t52
01/02/2019, 19:35:24	1	t53
01/02/2019, 22:42:21	3	t56
01/02/2019, 23:26:39	5	t57
01/03/2019, 00:43:45	2	t60
01/03/2019, 01:00:39	4	t61
01/03/2019, 02:37:05	1	t62
01/03/2019, 04:20:36	3	t65
01/03/2019, 06:22:42	2	t66
01/03/2019, 06:31:43	4	t67
01/03/2019, 06:49:39	5	t68
01/03/2019, 08:50:48	1	t71
01/03/2019, 10:13:17	3	t72
01/03/2019, 11:26:20	2	t73
⋮	⋮	⋮
01/03/2019, 12:37:06	4	t76
01/03/2019, 13:10:26	5	t77
01/03/2019, 15:17:31	1	t78
01/03/2019, 16:32:27	3	t81
01/03/2019, 16:56:58	2	t82
01/03/2019, 19:26:43	4	t83
01/03/2019, 20:09:19	1	t86
01/03/2019, 20:11:47	5	t87
01/03/2019, 22:08:42	2	t88
01/03/2019, 23:00:06	3	t89
01/03/2019, 23:29:27	4	t90
01/04/2019, 02:17:26	1	t93
01/04/2019, 02:21:47	5	t94
01/04/2019, 03:55:56	2	t95
01/04/2019, 04:47:09	3	t98
01/04/2019, 05:36:30	4	t99
01/04/2019, 09:35:35	2	t102
01/04/2019, 09:42:03	4	t103
01/04/2019, 09:46:30	1	t104
01/04/2019, 10:01:11	5	t105
01/04/2019, 11:46:30	3	t106
01/04/2019, 14:41:43	1	t109
01/04/2019, 14:46:40	4	t110
01/04/2019, 15:54:19	2	t111
01/04/2019, 16:41:16	5	t114
01/04/2019, 18:05:23	3	t115
01/04/2019, 18:43:03	1	t116
01/04/2019, 21:14:51	4	t119
01/04/2019, 22:27:28	5	t120
01/04/2019, 22:28:13	2	t121
01/04/2019, 22:58:27	1	t122
01/05/2019, 02:33:26	3	t125
01/05/2019, 03:06:12	4	t126
01/05/2019, 03:19:42	5	t127
01/05/2019, 04:19:23	2	t130
01/05/2019, 05:14:25	1	t131
01/05/2019, 07:02:56	3	t132
01/05/2019, 08:41:18	5	t135
01/05/2019, 08:42:11	4	t136
01/05/2019, 10:17:49	2	t137
01/05/2019, 13:37:30	1	t140
01/05/2019, 13:40:10	3	t141
01/05/2019, 14:24:26	4	t142
01/05/2019, 15:00:07	5	t143
01/05/2019, 16:35:10	2	t146
⋮	⋮	⋮
01/05/2019, 18:37:28	3	t147
01/05/2019, 19:46:34	5	t148
01/05/2019, 20:44:56	1	t151
01/05/2019, 21:15:19	4	t152
01/05/2019, 22:02:47	2	t153
01/05/2019, 23:49:35	3	t154
01/06/2019, 01:33:16	5	t157
01/06/2019, 03:57:44	4	t158
01/06/2019, 04:25:17	1	t161
01/06/2019, 04:45:16	2	t162
01/06/2019, 07:48:41	3	t163
01/06/2019, 08:53:40	5	t166
01/06/2019, 09:10:30	4	t167
01/06/2019, 10:31:14	1	t168
01/06/2019, 11:35:48	2	t169
01/06/2019, 14:55:25	5	t172
01/06/2019, 15:28:04	4	t173
01/06/2019, 15:33:23	3	t174
01/06/2019, 17:48:01	2	t177
01/06/2019, 17:55:07	1	t178
01/06/2019, 19:42:01	3	t179
01/06/2019, 20:59:49	4	t182
01/06/2019, 22:54:33	5	t183
01/06/2019, 23:38:50	1	t184
01/07/2019, 01:28:39	3	t187
01/07/2019, 02:11:42	2	t188
01/07/2019, 05:24:05	4	t191
01/07/2019, 06:15:38	1	t192
01/07/2019, 06:21:22	5	t193
01/07/2019, 07:24:41	3	t194
01/07/2019, 09:06:45	2	t197
01/07/2019, 10:39:43	4	t198
01/07/2019, 11:16:47	3	t199
01/07/2019, 11:59:36	1	t201
01/07/2019, 12:04:07	5	t203
01/07/2019, 15:00:00	2	t204
01/07/2019, 16:20:57	4	t207
01/07/2019, 16:42:33	5	t208
01/07/2019, 17:13:52	3	t209
01/07/2019, 17:26:38	1	t210
01/07/2019, 20:38:17	2	t213
01/07/2019, 22:00:01	4	t214
01/07/2019, 22:39:40	5	t215
01/07/2019, 23:34:36	1	t216
01/07/2019, 23:37:28	3	t217

Table 27: Base Case: Grid Times

Time	Grid balancing number	Annotation	length
01/01/2019, 04:00:00	1	t1	0 days 04:00:00
01/01/2019, 08:00:00	2	t7	0 days 04:00:00
01/01/2019, 12:00:00	3	t13	0 days 04:00:00
01/01/2019, 16:00:00	4	t16	0 days 04:00:00
01/01/2019, 20:00:00	5	t23	0 days 04:00:00
01/02/2019, 00:00:00	6	t28	0 days 04:00:00
01/02/2019, 04:00:00	7	t33	0 days 04:00:00
01/02/2019, 08:00:00	8	t40	0 days 04:00:00
01/02/2019, 12:00:00	9	t44	0 days 04:00:00
01/02/2019, 16:00:00	10	t49	0 days 04:00:00
01/02/2019, 20:00:00	11	t55	0 days 04:00:00
01/03/2019, 00:00:00	12	t59	0 days 04:00:00
01/03/2019, 04:00:00	13	t64	0 days 04:00:00
01/03/2019, 08:00:00	14	t70	0 days 04:00:00
01/03/2019, 12:00:00	15	t75	0 days 04:00:00
01/03/2019, 16:00:00	16	t80	0 days 04:00:00
01/03/2019, 20:00:00	17	t85	0 days 04:00:00
01/04/2019, 00:00:00	18	t92	0 days 04:00:00
01/04/2019, 04:00:00	19	t97	0 days 04:00:00
01/04/2019, 08:00:00	20	t101	0 days 04:00:00
01/04/2019, 12:00:00	21	t108	0 days 04:00:00
01/04/2019, 16:00:00	22	t113	0 days 04:00:00
01/04/2019, 20:00:00	23	t118	0 days 04:00:00
01/05/2019, 00:00:00	24	t124	0 days 04:00:00
01/05/2019, 04:00:00	25	t129	0 days 04:00:00
01/05/2019, 08:00:00	26	t134	0 days 04:00:00
01/05/2019, 12:00:00	27	t139	0 days 04:00:00
01/05/2019, 16:00:00	28	t145	0 days 04:00:00
01/05/2019, 20:00:00	29	t150	0 days 04:00:00
01/06/2019, 00:00:00	30	t156	0 days 04:00:00
01/06/2019, 04:00:00	31	t160	0 days 04:00:00
01/06/2019, 08:00:00	32	t165	0 days 04:00:00
01/06/2019, 12:00:00	33	t171	0 days 04:00:00
01/06/2019, 16:00:00	34	t176	0 days 04:00:00
01/06/2019, 20:00:00	35	t181	0 days 04:00:00
01/07/2019, 00:00:00	36	t186	0 days 04:00:00
01/07/2019, 04:00:00	37	t190	0 days 04:00:00
01/07/2019, 08:00:00	38	t196	0 days 04:00:00
01/07/2019, 12:00:00	39	t202	0 days 04:00:00
01/07/2019, 16:00:00	40	t206	0 days 04:00:00
01/07/2019, 20:00:00	41	t212	0 days 04:00:00

Table 28: Base Case: Δ_t in seconds

Time Annotation	Δ_t (s)
t0	0.000000
t1	77.552306
t2	3027.197531
t3	3306.764488
t4	3885.002009
t5	511.881988
t6	3609.153984
t7	60.000000
t8	10239.146457
t9	496.026079
t10	992.565444
t11	105.212805
t12	2507.049215
t13	60.000000
t14	3921.620709
t15	10418.379291
t16	60.000000
t17	4518.150088
t18	816.848502
t19	1691.897512
t20	1564.468429
t21	2545.705329
t22	3202.930140
t23	60.000000
t24	5891.295396
t25	7387.953025
t26	696.506127
t27	364.245452
t28	60.000000
t29	707.254418
t30	2601.254684
t31	7719.578192
t32	3311.912706
t33	60.000000
t34	5560.278134
t35	226.930425
t36	1897.162553
t37	4383.032274
t38	1407.072743
t39	865.523871
t40	60.000000
t41	11081.742667
t42	1088.613786
t43	2169.643547
t44	60.000000
t45	2883.578284
t46	6314.759648
t47	1526.477371
t48	3615.184697
t49	60.000000
t50	7448.488077
t51	84.920835
t52	114.592040
t53	5276.100566
t54	1415.898482
t55	60.000000
t56	9741.199247
t57	2658.786351
t58	1940.014402
t59	60.000000
t60	2625.321661
t61	1013.856196
t62	5786.710342
t63	4914.111801
t64	60.000000
t65	1236.202188
t66	7325.980664
t67	541.459705
t68	1075.979067
t69	4160.378376
t70	60.000000
t71	3048.956667
t72	4948.385033
t73	4382.881057
t74	1959.777243
t75	60.000000
t76	2226.157962
t77	1999.902877
t78	7625.391234
t79	2488.547927
t80	60.000000
t81	1947.024816
t82	1471.583960
t83	8984.744228
t84	1936.646996
t85	60.000000
t86	559.684110
t87	147.434715
t88	7014.961839
t89	3084.194598
t90	1760.834110
t91	1772.890628
t92	60.000000
t93	8246.176872
t94	261.458914
t95	5648.733379
t96	183.630835
t97	60.000000
t98	2829.431650
t99	2960.590102
t100	8549.978248
t101	60.000000
t102	5735.464279
t103	387.712676
t104	267.333124
t105	881.305555
t106	6318.276863
t107	749.907503
t108	60.000000
t109	9703.212490
t110	297.491112
t111	4058.572448
t112	280.723950
t113	60.000000
t114	2476.218226
t115	5046.987875
t116	2260.184579
t117	4556.609320
t118	60.000000
t119	4491.808765
t120	4356.601543
t121	45.415353
t122	1813.424754
t123	3632.749585
t124	60.000000
t125	9206.749904
t126	1965.338514
t127	809.937590
t128	2357.973992
t129	60.000000
t130	1163.112733
t131	3301.990272
t132	6511.689545
t133	3363.207450
t134	60.000000
t135	2478.160633
t136	53.429718
t137	5738.263169
t138	6070.146480
t139	60.000000
t140	5850.985396
t141	159.969541
t142	2655.093543
t143	2141.902031
t144	3532.049489
t145	60.000000
t146	2110.592654
t147	7337.885148
t148	4146.463178
t149	745.059020
t150	60.000000
t151	2696.273908
t152	1823.325564
t153	2847.428525
t154	6408.618061
t155	564.353942
t156	60.000000
t157	5596.485689
t158	8667.703420
t159	75.810891
t160	60.000000
t161	1517.713876
t162	1198.690193
t163	11005.546518
t164	618.049413
t165	60.000000
t166	3220.220405
t167	1009.800570
t168	4844.630078
t169	3873.762074
t170	1391.586873
t171	60.000000
t172	10525.735364
t173	1958.872220
t174	318.673937
t175	1536.718479
t176	60.000000
t177	6481.303997
t178	426.134014
t179	6413.575780
t180	1018.986209
t181	60.000000
t182	3589.701479
t183	6883.422571
t184	2657.821204
t185	1209.054746
t186	60.000000
t187	5319.028474
t188	2583.208717
t189	6437.762809
t190	60.000000
t191	5045.273660
t192	3093.206659
t193	344.439893
t194	3799.056010
t195	2058.023778
t196	60.000000
t197	4005.089366
t198	5578.500035
t199	2224.223230
t200	2532.187369
t201	36.138440
t202	23.861560
t203	247.454323
t204	10553.454147
t205	3539.091530
t206	60.000000
t207	1257.405875
t208	1296.530385
t209	1878.086530
t210	766.603199
t211	9141.374011
t212	60.000000
t213	2297.058860
t214	4904.372109
t215	2379.556520
t216	3295.388365
t217	171.675317
t218	1291.948829

Table 29: Grid Time Simulation

Time	Grid balancing number	Annotation	Length	Revenue
01/01/2019, 03:47:05	1	t0	0 days 04:21:39.911109	1.090208
01/01/2019, 08:09:44	2	t7	0 days 05:35:51.868707	1.399375
01/01/2019, 13:46:36	3	t14	0 days 02:51:36.825473	0.715000
01/01/2019, 16:39:13	4	t16	0 days 03:35:40.579391	0.898611
01/01/2019, 20:15:54	5	t23	0 days 04:18:56.459818	1.078889
01/02/2019, 00:35:50	6	t29	0 days 05:07:05.903423	1.279514
01/02/2019, 05:43:56	7	t35	0 days 04:29:38.571881	1.123472
01/02/2019, 10:14:35	8	t40	0 days 04:23:51.471508	1.099375
01/02/2019, 14:39:26	9	t46	0 days 05:19:40.377924	1.331944
01/02/2019, 20:00:06	10	t53	0 days 04:28:36.655142	1.119167
01/03/2019, 00:29:43	11	t57	0 days 03:44:46.784936	0.936528
01/03/2019, 04:15:30	12	t62	0 days 04:02:26.226541	1.010139
01/03/2019, 08:18:56	13	t68	0 days 03:16:34.542791	0.819028
01/03/2019, 11:36:31	14	t73	0 days 04:31:47.993415	1.132431
01/03/2019, 16:09:19	15	t78	0 days 04:55:33.001472	1.231458
01/03/2019, 21:05:52	16	t85	0 days 01:00:13.451624	0.250903
01/03/2019, 22:07:05	17	t87	0 days 05:12:05.335861	1.300347
01/04/2019, 03:20:10	18	t94	0 days 03:55:36.525397	0.981667
01/04/2019, 07:16:47	19	t99	0 days 04:26:39.198429	1.111042
01/04/2019, 11:44:26	20	t105	0 days 04:38:44.546442	1.161389
01/04/2019, 16:24:11	21	t111	0 days 01:41:17.628016	0.422014
01/04/2019, 18:06:28	22	t115	0 days 02:47:14.654479	0.696806
01/04/2019, 20:54:43	23	t118	0 days 03:39:00.835816	0.912500
01/05/2019, 00:34:44	24	t124	0 days 03:11:30.993225	0.797917
01/05/2019, 03:47:15	25	t129	0 days 04:15:27.022011	1.064375
01/05/2019, 08:03:42	26	t134	0 days 03:28:06.324279	0.867083
01/05/2019, 11:32:48	27	t139	0 days 02:16:20.143690	0.568056
01/05/2019, 13:50:08	28	t143	0 days 04:35:44.270325	1.148889
01/05/2019, 18:26:53	29	t148	0 days 05:05:00.465746	1.270833
01/05/2019, 23:32:53	30	t155	0 days 04:32:51.669095	1.136875
01/06/2019, 04:06:45	31	t160	0 days 03:16:13.922027	0.817569
01/06/2019, 07:23:59	32	t164	0 days 03:43:20.905877	0.930556
01/06/2019, 11:08:20	33	t170	0 days 04:31:54.341039	1.132917
01/06/2019, 15:41:14	34	t176	0 days 04:35:11.516581	1.146597
01/06/2019, 20:17:25	35	t181	0 days 04:59:24.700099	1.247500
01/07/2019, 01:17:50	36	t186	0 days 03:41:20.066373	0.922222
01/07/2019, 05:00:10	37	t190	0 days 02:33:44.830145	0.640556
01/07/2019, 07:34:55	38	t196	0 days 03:18:15.791611	0.826042
01/07/2019, 10:54:11	39	t200	0 days 04:31:01.985314	1.129236
01/07/2019, 15:26:13	40	t206	0 days 04:43:38.307854	1.181806

Table 30: Energy Consumption Scenarios (E_t)

Table 31: Standardized (BC)

Vessels	E^t
1	0.7
2	0.7
3	0.7
4	0.7
5	0.7
Grid Balancing	0.1

Table 32: Worst case

Vessels	E^t
1	0.9
2	0.9
3	0.9
4	0.9
5	0.9
Grid Balancing	0.6

Table 33: Random

Vessels	E^t
1	0.851
2	0.686
3	0.817
4	0.818
5	0.768
Grid Balancing	0.1

Table 34: Best case

Vessels	E^t
1	0.5
2	0.5
3	0.5
4	0.5
5	0.5
Grid Balancing	-0.4

Table 35: Grid balancing time simulation of two overlapping contracts

Time	Grid balancing number	Annotation	Length	Revenue
01/01/2019, 03:47:05	1	t0	0 days 04:21:39.911109	1.090208
01/01/2019, 06:20:00	42	t4	0 days 04:06:01.853113	1.025069
01/01/2019, 08:09:44	2	t8	0 days 05:35:51.868707	1.399375
01/01/2019, 10:27:02	43	t10	0 days 04:13:40.612273	1.056944
01/01/2019, 13:46:36	3	t17	0 days 02:51:36.825473	0.715000
01/01/2019, 14:41:43	44	t19	0 days 03:48:21.724238	0.951458
01/01/2019, 16:39:13	4	t21	0 days 03:35:40.579391	0.898611
01/01/2019, 18:31:05	45	t27	0 days 03:34:39.373700	0.894375
01/01/2019, 20:15:54	5	t30	0 days 04:18:56.459818	1.078889
01/01/2019, 22:06:44	46	t33	0 days 04:45:54.390685	1.191250
01/02/2019, 00:35:50	6	t38	0 days 05:07:05.903423	1.279514
01/02/2019, 02:53:38	47	t41	0 days 04:20:12.519410	1.084167
01/02/2019, 05:43:56	7	t46	0 days 04:29:38.571881	1.123472
01/02/2019, 07:14:51	48	t49	0 days 03:46:33.793851	0.943958
01/02/2019, 10:14:35	8	t53	0 days 03:42:03.182995	0.925208
01/02/2019, 11:02:25	49	t55	0 days 03:36:01.377931	0.900069
01/02/2019, 13:57:38	50	t60	0 days 03:51:36.948617	0.965000
01/02/2019, 14:39:26	9	t63	0 days 05:19:40.377924	1.331944
01/02/2019, 17:50:15	51	t66	0 days 03:35:32.464133	0.898056
01/02/2019, 20:00:06	10	t72	0 days 04:28:36.655142	1.119167
01/02/2019, 21:26:47	52	t74	0 days 05:14:18.625434	1.309583
01/03/2019, 00:29:43	11	t78	0 days 03:44:46.784936	0.936528
01/03/2019, 02:42:06	53	t83	0 days 04:55:49.653881	1.232569
01/03/2019, 04:15:30	12	t85	0 days 04:02:26.226541	1.010139
01/03/2019, 07:38:56	54	t91	0 days 02:57:39.681508	0.740208
01/03/2019, 08:18:56	13	t93	0 days 03:16:34.542791	0.819028
01/03/2019, 10:37:35	55	t97	0 days 05:09:01.806126	1.287569
01/03/2019, 11:36:31	14	t100	0 days 04:31:47.993415	1.132431
01/03/2019, 15:47:37	56	t105	0 days 03:24:40.883472	0.852778
01/03/2019, 16:09:19	15	t107	0 days 04:55:33.001472	1.231458
01/03/2019, 19:13:18	57	t111	0 days 02:52:47.259187	0.719931
01/03/2019, 21:05:52	16	t116	0 days 02:14:25.487426	0.560069
01/03/2019, 22:07:05	17	t118	0 days 05:12:05.335861	1.300347
01/03/2019, 23:21:17	58	t122	0 days 05:11:09.877946	1.296458
01/04/2019, 03:20:10	18	t127	0 days 03:55:36.525397	0.981667
01/04/2019, 04:33:27	59	t130	0 days 02:57:30.889319	0.739583
01/04/2019, 07:16:47	19	t134	0 days 04:23:48.378796	1.099167
01/04/2019, 07:31:58	60	t136	0 days 04:11:28.256620	1.047778
01/04/2019, 11:41:35	61	t142	0 days 02:51:10.288966	0.713194
01/04/2019, 11:44:26	20	t144	0 days 04:38:44.546442	1.161389
01/04/2019, 14:33:46	62	t147	0 days 02:02:31.321148	0.510486
01/04/2019, 16:24:11	21	t152	0 days 01:41:17.628016	0.422014
01/04/2019, 16:37:17	63	t154	0 days 04:16:26.038456	1.068472
01/04/2019, 18:06:28	22	t158	0 days 03:29:00.368164	0.870833
01/04/2019, 20:54:43	23	t161	0 days 03:39:00.835816	0.912500
01/04/2019, 21:36:29	64	t164	0 days 04:00:41.618522	1.002847
01/05/2019, 00:34:44	24	t169	0 days 03:11:30.993225	0.797917
01/05/2019, 01:38:10	65	t171	0 days 03:09:39.476648	0.790208
01/05/2019, 03:47:15	25	t176	0 days 03:25:23.393475	0.855764
01/05/2019, 04:48:50	66	t179	0 days 03:13:52.042197	0.807778
01/05/2019, 07:13:38	67	t183	0 days 03:34:54.697850	0.895417
01/05/2019, 08:03:42	26	t185	0 days 03:28:06.324279	0.867083
01/05/2019, 10:49:33	68	t190	0 days 02:59:35.398655	0.748264
01/05/2019, 11:32:48	27	t192	0 days 04:36:16.590458	1.151111
01/05/2019, 13:50:08	28	t196	0 days 04:35:44.270325	1.148889
01/05/2019, 16:10:05	69	t200	0 days 04:23:08.888737	1.096389
01/05/2019, 18:26:53	29	t203	0 days 05:05:00.465746	1.270833
01/05/2019, 20:34:14	70	t207	0 days 03:36:38.058901	0.902639
01/05/2019, 23:32:53	30	t212	0 days 03:57:39.373964	0.990208
01/06/2019, 00:11:52	71	t215	0 days 03:53:53.010760	0.974514
01/06/2019, 03:31:32	72	t218	0 days 03:51:26.217158	0.964306
01/06/2019, 04:06:45	31	t221	0 days 03:30:08.628367	0.875556
01/06/2019, 07:23:59	32	t225	0 days 02:40:04.406933	0.666944
01/06/2019, 07:37:53	73	t227	0 days 03:29:26.199537	0.872639
01/06/2019, 10:05:03	74	t232	0 days 05:19:41.303438	1.332014
01/06/2019, 11:08:20	33	t235	0 days 04:31:54.341039	1.132917
01/06/2019, 15:25:44	75	t239	0 days 04:29:11.481054	1.121597
01/06/2019, 15:41:14	34	t243	0 days 04:35:11.516581	1.146597
01/06/2019, 19:55:56	76	t248	0 days 05:10:39.662169	1.294375
01/06/2019, 20:17:25	35	t250	0 days 04:59:24.700099	1.247500
01/07/2019, 01:07:36	77	t255	0 days 03:25:11.379584	0.854931
01/07/2019, 01:17:50	36	t257	0 days 03:41:20.066373	0.922222
01/07/2019, 04:33:47	78	t261	0 days 03:00:08.126936	0.750556
01/07/2019, 05:00:10	37	t263	0 days 04:23:54.086732	1.099583
01/07/2019, 07:34:55	38	t269	0 days 03:18:15.791611	0.826042
01/07/2019, 09:25:04	79	t272	0 days 04:40:16.457012	1.167778
01/07/2019, 10:54:11	39	t275	0 days 04:31:01.985314	1.129236
01/07/2019, 14:06:21	80	t280	0 days 02:27:00.952614	0.612500
01/07/2019, 15:26:13	40	t283	0 days 04:43:38.307854	1.181806
01/07/2019, 16:34:22	81	t286	0 days 04:07:56.258524	1.033056

Table 36: Case 2 simulated E_t uniformly distributed under Assumption 6.2.

Time Annotation	Energy Consumption E_t
t0	0.728561
t1	0.028889
t2	0.731237
t3	0.582439
t4	0.825329
t5	0.829436
t6	0.000000
t7	0.253473
t8	0.564092
t9	0.708268
t10	0.631109
t11	0.599999
t12	0.000000
t13	0.552817
t14	0.898623
t15	0.000000
t16	-0.355444
t17	0.844064
t18	0.741276
t19	0.652642
t20	0.613447
t21	0.769986
t22	0.000000
t23	0.056831
t24	0.774345
t25	0.764739
t26	0.553191
t27	0.000000
t28	0.367838
t29	0.892965
t30	0.887755
t31	0.745331
t32	0.000000
t33	-0.355739
t34	0.501622
t35	0.553589
t36	0.876401
t37	0.621144
t38	0.646458
t39	0.000000
t40	0.498196
t41	0.625746
t42	0.719593
t43	0.000000
t44	0.036031
t45	0.525998
t46	0.733818
t47	0.837627
t48	0.000000
t49	-0.243581
t50	0.589720
t51	0.665148
t52	0.514770
t53	0.698641
t54	0.000000
t55	0.417983
t56	0.763156
t57	0.713393
t58	0.000000
t59	0.455126
t60	0.559875
t61	0.726894
t62	0.649670
t63	0.000000
t64	0.201305
t65	0.545166
t66	0.810205
t67	0.538642
t68	0.566545
t69	0.000000
t70	0.407450
t71	0.879078
t72	0.673293
t73	0.665632
t74	0.000000
t75	-0.154518
t76	0.609881
t77	0.746901
t78	0.571384
t79	0.000000
t80	-0.280517
t81	0.682294
t82	0.565975
t83	0.759490
t84	0.000000
t85	0.421056
t86	0.811005
t87	0.692053
t88	0.639029
t89	0.673884
t90	0.502003
t91	0.000000
t92	0.312573
t93	0.632663
t94	0.627769
t95	0.531970
t96	0.000000
t97	0.048056
t98	0.733140
t99	0.656235
t100	0.000000
t101	0.469546
t102	0.769341
t103	0.596563
t104	0.710142
t105	0.864214
t106	0.708044
t107	0.000000
t108	0.202105
t109	0.525109
t110	0.695754
t111	0.684577
t112	0.000000
t113	0.001130
t114	0.668174
t115	0.733974
t116	0.715467
t117	0.000000
t118	0.089896
t119	0.566319
t120	0.676539
t121	0.887471
t122	0.666069
t123	0.000000
t124	-0.364682
t125	0.501132
t126	0.714167
t127	0.520229
t128	0.000000
t129	-0.307257
t130	0.543503
t131	0.681631
t132	0.897075
t133	0.000000
t134	0.085055
t135	0.683456
t136	0.674604
t137	0.700445
t138	0.000000
t139	0.047087
t140	0.790486
t141	0.859177
t142	0.779148
t143	0.669324
t144	0.000000
t145	0.249529
t146	0.863816
t147	0.561451
t148	0.594937
t149	0.000000
t150	0.219112
t151	0.774808
t152	0.757040
t153	0.735411
t154	0.831769
t155	0.000000
t156	0.078943
t157	0.823036
t158	0.884942
t159	0.000000
t160	0.035357
t161	0.843538
t162	0.787297
t163	0.855473
t164	0.000000
t165	-0.353764
t166	0.865239
t167	0.806735
t168	0.884952
t169	0.892528
t170	0.000000
t171	-0.114514
t172	0.768001
t173	0.556509
t174	0.738053
t175	0.000000
t176	0.519580
t177	0.576789
t178	0.639947
t179	0.531898
t180	0.000000
t181	-0.205344
t182	0.583081
t183	0.538400
t184	0.899879
t185	0.000000
t186	0.487534
t187	0.685504
t188	0.797481
t189	0.000000
t190	0.295237
t191	0.708982
t192	0.669395
t193	0.765629
t194	0.722208
t195	0.000000
t196	-0.134110
t197	0.645730
t198	0.750907
t199	0.750852
t200	0.000000
t201	0.583838
t202	0.209773
t203	0.849972
t204	0.848535
t205	0.000000
t206	0.252263
t207	0.528069
t208	0.669407
t209	0.535609
t210	0.522100
t211	0.000000
t212	-0.145302
t213	0.682750
t214	0.803760
t215	0.751537
t216	0.612994
t217	0.778028
t218	0.000000

E Appendix Lookup Lists

List of Theorems

List of Lemmas

List of Assumptions

List of Algorithms

List of Definitions

List of Figures

List of Tables