NTNU
Norwegian University of
Science and Technology

Heuristics for a Dual-Level Stochastic Fleet Size and Mix Problem for Maintenance Operations at Offshore Wind Farms

Kamilla Hamre Bolstad
Manu Joshi

Industrial Economics and Technology Management
Submission date: June 2016
Supervisor: Marielle Christiansen, IØT
Co-supervisor: Magnus Stålhane, IØT
Lars Magnus Hvattum, Høgskolen i Molde

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

Problem Description

The purpose of this thesis is to develop model(s) and solution method(s) to determine an optimal fleet size and mix to be used in the execution of maintenance operations at offshore wind farms. The problem is studied from a strategic point of view, where the aim is to make optimal long-term strategic decisions minimizing maintenance costs. In this thesis a dual-level stochastic approach is used to be able to account for both long-term strategic uncertainty and short-term tactical uncertainty, combining decisions with different time scales in one optimization model. A first version of a dual-level stochastic model and a scenario generator was developed in our project report. In this master thesis, the model and scenario generator will be analyzed and improved, and methods for solving the problem will be explored. Heuristic solution method(s) will be developed for the model, and computational experiments will be conducted to evaluate the performance of exact and heuristic solution methods.

Preface

This master thesis is the concluding part of our Master of Science at the Department of Industrial Economics and Technology Management, at the Norwegian University of Science and Technology (NTNU). The thesis is written within the field of Applied Economics and Operation Research, and studies a strategic fleet size and mix problem for conducting maintenance operations at offshore wind farms. The thesis is a continuation of the work conducted for our specialisation project during the fall semester 2015.

We would like to give a special thanks to our supervisors Professor Lars Magnus Hvattum (Faculty of Logistics, Molde University College), Associate Professor Magnus Stålhane (Department of Industrial Economics and Technology Management, NTNU) and Professor Marielle Christiansen (Department of Industrial Economics and Technology Management, NTNU) for their thorough and valuable guidance throughout the semester.

Abstract

Due to high costs, the offshore wind industry is currently not economically viable on its own. For offshore wind to be a profitable future energy source, it is therefore necessary to find methods for reducing costs. Operations and maintenance at offshore wind farms is an important cost driver, and accounts for 20 - 25% of the lifetime costs of wind farms today. Vessel fleets used to conduct maintenance operations at offshore wind farms are expensive to acquire and operate. A vessel fleet that is capable of operating in rough weather conditions gives a high degree of accessibility to wind farms, and hence allows wind farm owners to reduce revenue loss from unplanned production stops. The potential savings from determining an optimal fleet size and mix for conducting maintenance at offshore wind farms can therefore be substantial.

This thesis studies the strategic problem of finding a cost optimal fleet size and mix for conducting maintenance operations at offshore wind farms (DLPOW). A dual-level stochastic model has been developed, which accounts for both long-term strategic uncertainty and short-term tactical uncertainty in one optimization model. The model supports wind farm owners in making strategic decisions regarding the amount, placement, charter length and types of vessels to long- and short-term charter, to meet maintenance demand throughout the lifetime of a wind farm. To evaluate the quality of strategic fleet size and mix decisions, the model also considers the tactical deployment decisions of how to utilize the fleet to conduct maintenance operations. The model accounts for strategic uncertainties that have not been considered in previously developed optimization models for offshore wind, such as uncertainty related to: long-term trends in electricity prices and subsidy levels, stepwise development of wind farms, and technology development in the vessel industry.

Several solution methods have been developed to solve the DLPOW. From computational testing, it can be seen that the use of a standard optimization solver as a solution method is impractical for anything but small instances due to memory limitations. In order to solve real-life instances, a heuristic solution method based on the Greedy Randomized Adaptive Search Procedure (GRASP) has been developed. The reactive metaheuristic developed exploits the block-separable structure of the DLPOW to decompose the problem into a master problem and many independent subproblems. The reactive GRASP constructs strategic fleet size and mix solutions for solving the master problem. A simple Greedy Tactical Heuristic, embedded in the GRASP, solves the subproblems of tactical fleet deployment to evaluate the objective function value of a given fleet solution. The performance of the reactive GRASP has been evaluated by comparing solution time and quality from the GRASP to the equivalent values obtained from a standard optimization solver. The results show that the GRASP consistently provides good solutions for the DLPOW, and finds higher quality solutions for 24 of 29 tested instances, compared to the solver. Furthermore, the GRASP is able to solve significantly larger problems within a considerable shorter amount of time.

Sammendrag

Som følge av høye kostnader, er offshore vindindustri per i dag ikke levedyktig på egenhånd. For at offshore vindkraft skal kunne bli en lønnsom energikilde i fremtiden, er det nødvendig å finne metoder for å redusere kostnader. Drift og vedlikehold av offshore vindparker er en viktig kostnadsdriver, og utgjør 20 - 25% av livsløpskostnadene til dagens vindparker. Fartøysflåten som brukes til å utføre vedlikeholdsaktiviteter er kostbar å anskaffe og operere. Flåter som kan operere i krevende værforhold gir en høy grad av tilgjengelighet til vindparker, og gjør dermed at eierne av vindparkene kan redusere inntektstap fra uventede produksjonsstopp. De potensielle besparelsene fra å bestemme en optimal flåtestørrelse og flåtemiks for å utføre vedlikehold på offshore vindparker, kan derfor ansees som betydelige.

Denne avhandlingen studerer det strategiske problemet som omhandler å finne en kostnadsoptimal flåtestørrelse og flåtemiks for å utføre vedlikehold på offshore vindparker (DLPOW). En dual-level stokastisk modell, som tar høyde for både langsiktig strategisk usikkerhet og kortsiktig taktisk usikkerhet i én optimeringsmodell, har blitt utviklet. Modellen støtter eiere av vindparker i strategisk beslutningstaking relatert til mengde, plassering, leielengde og type fartøy som bør lang- og korttidsleies, for å møte etterspørsel av vedlikehold gjennom levetiden til en vindpark. For å evaluere kvaliteten av strategiske beslutninger relatert til flåtestørrelse og flåtemiks, tar modellen også høyde for taktiske beslutninger om hvordan flåten skal brukes til å gjøre vedlikehold. Modellen tar høyde for strategisk usikkerhet som ikke har blitt tatt høyde for i tidligere utviklede optimeringsmodeller for offshore vindindustri, som usikkerhet relatert til: langsiktige trender i elektrisitetspriser og subsidier, stegvis utvikling av vindparker, og teknologiutvikling i skipsindustrien.

Flere løsningsmetoder har blitt utviklet for å løse DLPOW. Resultater fra beregningsorienterte tester, viser at det å bruke et standard optimeringsverktøy som løsningsmetode er upraktisk for alt annet enn små instanser på grunn av minnebegrensinger. For å løse instanser av realistisk størrelse, har en heuristisk løsningsmetode basert på prosedyren Grådig Randomisert Adaptiv Søkeprosedyre (GRASP) blitt utviklet. Den reaktive metaheuristikken som har blitt utviklet, utnytter den blokk-separable strukturen til DLPOW for å dekomponere problemet til et masterproblem og mange uavhengige subproblemer. Den reaktive GRASPen konstruerer strategiske løsninger for masterproblemet. En enkel grådig taktisk heuristikk er innebygd i GRASPen for å løse de taktiske subproblemene relatert til bruk av flåte, og evaluerer objektivfunksjonsverdien av en gitt flåteløsning. Prestasjonsevnen til den reaktive GRASPen har blitt evaluert ved å sammenligne løsningstid og løsningskvalitet fra GRASPen med ekvivalente verdier fra et standard optimeringsverktøy. Resultatene viser at GRASPen gir konsistent gode løsninger på DLPOW, og sammenlignet med optimeringsverktøyet finner GRASPen løsninger av høyere kvalitet for 24 av 29 instanser. Videre klarer GRASPen å løse signifikant større problemer innen betraktelig kortere tid.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The world energy demand is increasing, and the EU has set ambitious targets for energy consumption from renewable sources by 2020 [32]. As a green and renewable energy source, power generated from offshore wind plays an important part in reaching these targets. The wind industry has been growing steadily the last years, and power from wind constituted 15.6% of the EU power mix in 2015. Today, offshore wind only accounts for a small part of the total wind energy produced. However, the offshore wind industry is in rapid growth, and it is expected that energy from offshore wind will account for 7.7% of the EU power mix in 2030.

In recent years, wind power generation has been driven offshore for several reasons. Space is one of the most important contributors, as appropriate locations for onshore wind farms is becoming a scarce resource. Greater areas offshore allows larger wind farms to be built, and together with the reduction of noise and visual impact on the coast, moving wind farms offshore allows the use of larger turbine designs which improve efficiency [33]. Furthermore, the high wind speeds offshore makes the turbines yield a higher total electricity production. Where an onshore turbine would generate 2000 - 2500 full load hours per year, the comparable number for an offshore turbine could be up to 4000 [82]. This makes offshore wind production very attractive, resulting in investments and growth in the industry.

According to the Global Wind Energy Council (GWEC), offshore wind power has the potential to meet Europe's energy demand 7 times over [61]. However, while electricity from onshore wind farms is becoming cheaper than conventional power in an increasing number of markets, high costs are still a big challenge for offshore wind [61]. In general, offshore wind projects are considered to be around 50% more expensive than onshore wind projects. Due to the high expenses, offshore wind is still not profitable on its own and depends on governmental subsidies. Operations and maintenance (O&M) costs account for a substantial amount of the expenses, and is estimated to be around €0.012 - 0.015 per kWh of wind power produced [82]. This constitutes 20 - 25% of the total lifetime cost of an offshore wind farm [109]. In order to make offshore wind power profitable and viable without governmental subsidies, finding ways to minimize the O&M costs is hence of crucial importance.

The main cost components of maintenance operations are related to the costs of acquiring and operating a vessel fleet to conduct maintenance, and the loss of revenue incurred when turbines are shut down due to failures or maintenance execution (downtime cost). In general, studies show that the cost of acquiring and operating a vessel fleet accounts for up to 73% of O&M costs [39], while the loss of revenue during downtimes accounts for up to 66% of O&M costs [26]. The high maintenance cost of offshore wind is to a large extent caused by the rough weather conditions offshore, which makes the turbines more exposed to breakdowns and more difficult to access. Difficulties in accessing wind farms to repair failures may lead to long periods of downtime for the turbines, resulting in significant loss of revenue. The operational capabilities of the vessels used to conduct maintenance is therefore important, as the vessels ability to handle harsh weather conditions directly influence the accessibility of the wind farm.

When composing a vessel fleet for conducting maintenance, different strategic decisions need to be made, which all have consequences on the costs of conducting O&M. Firstly, a choice of which vessel concepts to acquire must be made. The characteristics of the different vessel concepts in the fleet affect the accessibility and travel time to the wind farm, and the ability to perform maintenance tasks. Secondly, a choice regarding the method and timing of acquisition needs to be considered. Different charter contracts have different costs, and the charter rates of vessels can vary significantly from year to year, and between vessel concepts. Lastly, the decision of where to locate the maintenance vessels also needs to be considered. Locating vessels offshore in close proximity to the wind farms can reduce travel times and increase farm accessibility significantly. However, offshore stations are expensive to install and maintain. Due to the large number of decisions and possible choices that have to be made, finding an optimal fleet size and mix which minimizes O&M costs is not easy.

To complicate further, the strategic fleet size and mix decisions place restrictions on the tactical decisions of how to deploy the fleet to conduct maintenance. In order to evaluate the quality of fleet size and mix decisions, the cost of optimal deployment hence must be considered. Furthermore, both the strategic fleet size and mix decision and the tactical deployment decision are subject to a wide range of uncertainties. At the strategic level, examples of such uncertainties are: long-term trends in electricity prices, the level of future governmental subsidies, the introduction of new vessel concepts in the market, and whether or not new turbines will be added to a wind farm in the future. At the tactical level, uncertain parameters like weather conditions and demand for maintenance influence the decision of how to utilize the fleet significantly. Considering the high costs related to conducting O&M, and the complexity of finding an optimal fleet size and mix, the use of operations research (OR) can provide wind farm owners with support to make better and more informed decisions.

In this thesis, the Dual-Level fleet size and mix Problem for conducting maintenance at Offshore Wind farms (DLPOW) is studied. An OR approach is used to develop a mathematical model for minimizing the costs of conducting maintenance at one or several wind farms. The model aims to find an optimal vessel fleet size and mix at the beginning of a wind farm's lifetime, while considering the optimal deployment of the fleet and the possibility of periodic fleet adjustments throughout the wind farms lifetime. In order to capture uncertainty at both strategic and tactical planning level, a dual-level stochastic modelling approach is used. The work presented in thesis is a continuation of the authors specialisation project, presented in [23], where a first version of the dual-level stochastic model was developed. In order to solve real-life instances of the DLPOW, heuristic solution methods for the problem are studied in this thesis. A simple heuristic solution method is implemented trough the use of a standard optimization solver. Moreover, a heuristic solution method, based on the metaheuristic Greedy Randomized Adaptive Search Procedure (GRASP), is developed. The GRASP constructs fleet size and mix solutions to the DLPOW in an iterative greedy manner, by utilizing a simple greedy heuristic for evaluating the cost of a given fleet. The heuristic solution methods are extensively tested in this thesis, and their performance is compared to an exact solution method.

Offshore wind is a relatively young industry, and the number of publications studying the fleet size and mix problem for this industry is limited. To the authors knowledge, previous work on the fleet size and mix problem for offshore wind have only accounted for uncertainty at the tactical level. The model developed in this thesis hence represents a new approach to the strategic fleet size and mix problem in offshore wind, as it considers decisions made on two different time scales, and accounts for uncertainty at both the strategic and the tactical planning level. To the authors knowledge, GRASP has never been applied on a fleet size and mix problem in any industry. Furthermore, the metaheuristic developed in this thesis is the first application of GRASP on a dual-level stochastic model, and possibly the second application on a stochastic model in general [74].

The thesis is organised as follows. In Chapter 2, relevant background information regarding O&M at offshore wind farms is presented. A thorough description of the problem studied in this thesis is given in Chapter 3, and relevant literature is reviewed in Chapter 4. A mathematical formulation of the problem, and the underlying model assumptions, are described in Chapter 5. Furthermore, the GRASP developed to solve the DLPOW is presented in Chapter 6. In order to generate test instances for the solution methods, a scenario generator has been developed. The scenario generator and the choice of critical input parameters are described in Chapter 7. In Chapter 8, computational experiments conducted to test the model, the scenario generator and the solution methods are described. Concluding remarks and suggestions for further research is presented in Chapter 9.

Chapter 2

Background

Maintenance Operations at offshore wind farms deal with challenges from two merging industries: offshore wind and maritime transportation. In this chapter, important aspects from these two industries are presented in order to give the reader a thorough understanding of the problem studied.

2.1 The Energy Sector

The emission of greenhouse gasses is affecting the climate of the earth [76]. The energy sector is facing enormous challenges both in terms of reducing the emission of greenhouse gasses, and at the same time meeting the ever-increasing demand for energy. The 21st Conference of the Parties of the UNFCCC was held in Paris in December 2015, where the aim was to adopt a new global agreement limiting the emission of greenhouse gasses [76]. As the global energy production and use currently accounts for 2/3 of the emission, a large part of the global agreement includes policies for transforming the energy sector and using more renewable energy sources [76].

According to the International Energy Agency, the world energy demand is expected to increase with 37% by 2040 [76]. Furthermore, the legally binding target for renewable energy by 2030, set by the European Union (EU) in 2014, states that at least 27% of the final energy consumption at European level should come from renewable energy sources [32]. This translates into 46 - 49% of electricity generated by renewables. The European Wind Energy Association (EWEA) expects wind energy to take the bigger share, of 21%, of this [32].

The wind industry is in rapid growth. As illustrated in Figure 2.1, wind power has increased its share of the total installed power capacity in EU with a factor of 6.5 times since year 2000, contributing 15.6% in 2015 [35]. Wind energy has hence overtaken hydro as the third largest power generation capacity in the EU, and is the largest source of renewable energy[35].

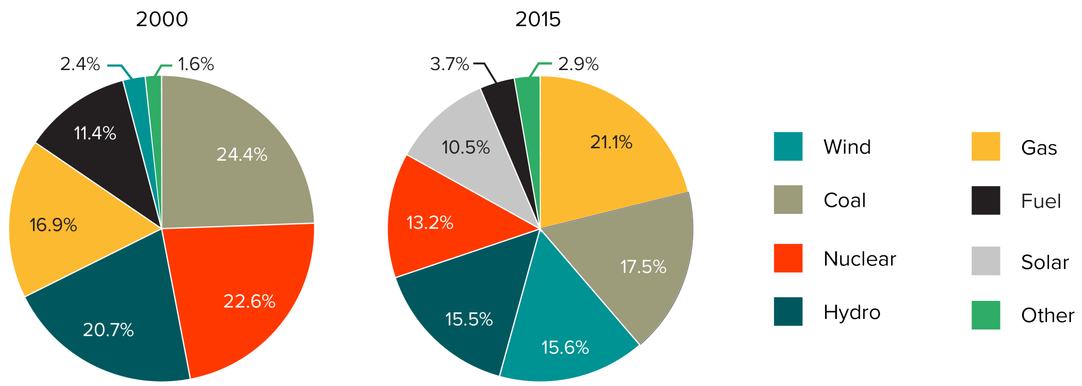


Figure 2.1: EU power mix in 2000 and 2015, retrieved from [35].

In 2015 the total installed wind power capacity was 141.6 GW in EU and 432.4 GW globally [60]. In EU, the annual installations of wind power has increased from 3.2 GW in 2000 to 12.8 GW in 2015 [35]. Globally, annually installed wind power has also grown steadily from from 1997, with a total of 61.0 GW installed in 2015 [60]. Wind energy is expected to continue to grow, and play an important role in the future energy market [60]. EWEA presents three scenarios for 2030 on how wind energy can contribute to meeting future electricity demand in EU. These scenarios are illustrated in Figure 2.2. In EWEA's central scenario, it is expected that wind energy will account for 24.4% of EU's electricity demand, with a total of 320 GW installed capacity [32].

Before the early 2000's, virtually all wind power produced was generated by onshore wind farms [3], and offshore wind still only accounts for a small fraction of the total installed wind capacity worldwide. In 2014, the share of EU consumption met by offshore wind was 1.1%, and 10.4% of European wind energy production was generated offshore [34]. However, offshore wind is expected to meet 7.7% of EU electricity demand in EWEA's central scenario for 2030, accounting for 31.6% of the total EU demand met by wind energy, as shown in Figure 2.2. In the past 15 years, offshore wind has experienced a steady growth, and as of February 2016 there are 3,230 offshore wind turbines with a combined capacity of 11.0 GW connected to the European grid [36]. Europe dominates the offshore wind segment, as more than 95% of global operational offshore installations are located in European waters [82].

	Installations (GW)			Generation (TWh)			EU electricity demand met by wind energy (%)		
	Onshore	Offshore	Total	Onshore	Offshore	Total	Onshore	Offshore	Total
Low Scenario	206.3	44.6	250.9	440.2	164.2	604.5	13.8%	5.2%	19%
Central Scenario	253.6	66.5	320.1	533.1	244.5	777.7	16.7%	7.7%	24.4%
High Scenario	294.0	98.1	392.1	627.5	360.8	988.3	19.7%	11.3%	31%

Figure 2.2: EWEA's 2030 scenarios for wind energy, retrieved from [32].

2.2 Trends in the Offshore Wind Industry

The offshore wind industry is relatively young. The first offshore wind turbine was installed in Sweden in 1990. The construction consisted of a single 220 kW turbine located 350 m from the coast at about 6 m depth [47]. Since then, the offshore wind industry has experienced a rapid growth and significant technological developments, leading to larger wind farms, increased turbine and more flexibility in the location of the wind farms.

Due to national maritime spatial planning and wind farm developers' desire to harness better energy resources out at sea, wind farms are built further and further from the coast. In 2002, the average European offshore wind turbine was placed 9.8 km from shore [56]. In comparison, the average distance from shore was 43.3 km in 2015 [36]. The German wind farm Bard Offshore 1 is one of the wind farms placed furthest from shore with a distance of 100 km [138]. Placement further offshore is advantageous due to higher wind speed and increased wind stability, as well as the avoidance of visual impact on the coast. However, larger distances from shore make O&M activities even more challenging, as transportation times increase and rougher weather conditions increase the risk of turbines being inaccessible for periods of time.

As wind farms are placed further from shore, the water depth also generally increases. In 2002, the average depth of operating European wind farms was 6.2 m [56], while in 2015 the average depth was 27.1 m [36]. As the substructures of commercial turbines are attached to the seabed, the depth of the location is a restricting factor, and current substructures are economically limited to water depths of 40 to 50 m [33]. However, since deep-water offshore designs have a great potential for unlocking new promising locations, extensive R&D efforts have been placed on developing new technologies to overcome this challenge. An example of such technology is the floating turbine, which has a floating substructure that is not attached seabed. Currently there are two operative full-scale floating turbines, Hywind and Windfloat, in use in Europe [33].

As far-shore wind farms generally are more expensive to install and operate, they need larger wind turbines and higher energy output to balance costs and revenue. This has led to development of new turbine technology, with considerable increments in turbine size and capacity to increase energy yields at sea. At the moment, wind turbines with capacities of up to 7 MW are being tested [46]. In 2015, the average capacity of new offshore wind turbines installed in Europe was 4.2 MW, a significant increase from 3.0 MW in 2010 [36]. The size of wind farms (given by the total installed capacity) is also rapidly increasing, leading to benefits from economies of scale. In 2010, the average size of offshore wind farms in Europe was 155.3 MW, while in 2015 the average was 337.9 MW [36]. The biggest operating offshore wind farm in the world today is London Array located 20 km offshore Kent in the UK. The wind farm has a total capacity of 630 MW and consists of 175 turbines. The plan is to expand the wind farm further through a second installation phase giving an installed capacity of 1 GW [87].

The mentioned trends are expected to continue, leading to even larger wind farms placed further from shore in deeper waters. The trends can be recognized in current projects under construction, like Forewind's Dogger Bank. The Dogger Bank project encompasses four 1.2 GW wind farms, each with 200 turbines, covering an area of 8660 km^2 [55]. The turbines are located 125 - 290 km offshore the UK in water depths ranging from 18 - 63 m [55]. The project is developed in stages, and if fully realized, it will be the largest offshore wind farm in the world. Projects like the Dogger Bank illustrate another implication of recent trends: as the projects grow larger and move further offshore, the need for stepwise installments increase as it takes several years to complete the installment.

2.3 O&M at Offshore Wind Farms

The cost of O&M activities constitute 20 - 25% of the total lifetime costs of an offshore wind farm [109]. As the name implies, O&M comprises two different streams of activities conducted during the operational phase of a wind farm. Operations refers to the high-level management of assets, like remote and environmental monitoring, electricity sales, marketing and administration [109]. Maintenance is the up-keep and repair of the physical farm and its systems, and comprises maintenance tasks on various equipment like turbines, export cable and grid connection, array cables and turbine foundations. Operations represents a very small proportion of O&M expenditure, while maintenance accounts for the largest portion of O&M effort, cost and risk [109].

The availability of a wind farm is considered an important measure of the performance of O&M activities. Availability is defined as the proportion of the time that a turbine, or the wind farm as a whole, is technically capable of producing electricity [109]. To ensure a satisfactory level of availability of the turbines, maintenance is required. The maintenance tasks can be divided in two main categories: preventive (scheduled) and corrective (unscheduled) maintenance. Preventive maintenance is done precautionary to prevent failures, while corrective maintenance tasks are performed when an unexpected failure has occurred.

2.3.1 Preventive Maintenance

Preventive maintenance is conducted with the intention of expanding a turbine's lifetime, and keeping the need for corrective maintenance at a reasonable level. Preventive maintenance includes inspections, testing, and maintenance activities like: oil sampling, pitch calibration, re-tightening of bolts and change of consumables. The frequency of preventive maintenance vary, and depends on the maintenance strategy of the wind farm owner. Four different maintenance strategies can be identified: no-maintenance, corrective maintenance only, opportunity maintenance and periodic maintenance [57].

In a no-maintenance strategy only major overhauls every five years are conducted. Given current levels of turbine failure rates, this strategy is however not a viable option [57]. In a corrective maintenance only strategy, maintenance is only performed when failures occur, and no preventive maintenance is conducted. If an opportunity-based strategy is used, corrective maintenance is executed on demand and preventive maintenance is performed at the same time. The most commonly used strategy is periodic maintenance. With this strategy preventive maintenance tasks are scheduled and corrective tasks are conducted when needed. The frequency of preventive maintenance affects the costs of O&M. To minimize costs, a balance must be found between conducting maintenance activities too often and too seldom [124]. Under-maintenance increases the risk of failures, and thereby the expected loss of revenue due to production stops. Over-maintenance leads to reduction in incremental benefits, waste of resources, and increased maintenance costs.

In a periodic maintenance strategy, preventive maintenance is usually performed once or twice a year, and several maintenance tasks are accumulated and performed in one or several visits to each turbine. Normally, preventive maintenance is scheduled to be executed during summer, as accessibility is higher due to better weather conditions. Furthermore, average wind speeds are generally lower during summer leading to a lower impact on production. While conducting certain types of preventive maintenance tasks, the turbine is shut down upon arrival and restarted when the task is completed. For other types of tasks, the turbine can continue running while maintenance is performed. The downtime of the turbine is therefore limited to the time it takes to perform the maintenance task.

2.3.2 Corrective Maintenance

Corrective maintenance are reactive activities conducted due to unexpected failures, and typically involves the repair or replacement of failed or damaged components [109]. When an unexpected failure occurs, the turbine is shut down until the corrective maintenance task has been conducted. When a failure leads to unscheduled production stops, an additional cost of lost revenue from electricity sales is introduced, called downtime cost. To reduce the downtime costs, corrective maintenance should preferably be performed immediately after a failure has occurred. However, this is not always feasible due to rough weather conditions and low accessibility to the wind farm. Furthermore, the repair of a turbine may be delayed if the part that needs replacement is out of stock.

The frequency and timing of turbine failure is subject to significant uncertainty, but is possible to forecast based on historical data. As the offshore wind industry is relatively young, such failure data is scarce. However, several studies have been conducted on this matter for onshore wind turbines. Reliability specialists suggest that the failure rates of a turbine typically follows a bathtub curve, indicating that most failures occur in the beginning and end of a turbine's lifetime [70]. In the middle of its life-time, the turbine experiences relatively low and constant failure rates. This is shown in Figure 2.3.

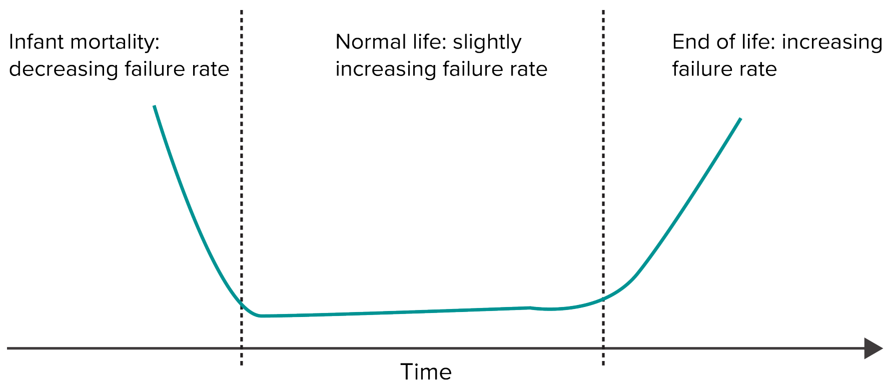


Figure 2.3: The bathtub curve of wind turbine failure rates, based on [70].

Wind turbines are complex machines functioning in complex environments. They are built by the integration of various technologies, and several components can be the source of failure. A study conducted by the German Wind Energy Measurement Programme, gathered statistics on the probabilities of component failures and related turbine downtime. The study tracked the performance of 1500 onshore wind turbines in Germany for 10 years, from 1997 to 2006 [96]. The results from the study is shown in Figure 2.4. As the data is based on onshore turbines, it may not be representative for offshore turbines, which are exposed to rougher conditions and have lower accessibility. However, the data may serve as a starting point when estimating turbine failure rates for offshore turbines.

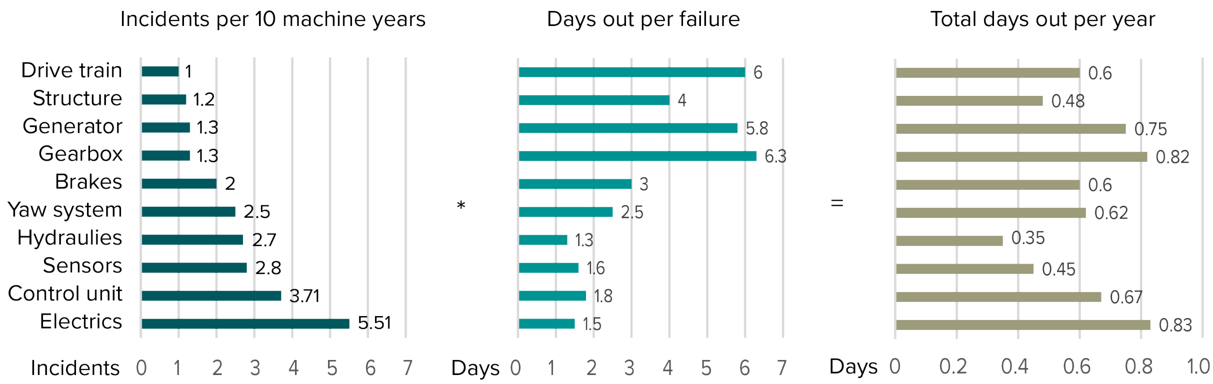


Figure 2.4: Failure rates of onshore wind turbine components, based on [96].

Another approach to forecasting failure rates for offshore turbines, is to base the forecast on expert knowledge. Such an approach is used by Dinwoodie in [43]. Failures are categorised into 5 types according to the maintenance impact of the fault, ranging from last sever to most significant: 1) manual reset, 2) minor repair, 3) medium repair, 4) major repair and 5) major replacement. Each failure type is given an average annual failure rate based on expert knowledge from turbine developers.

2.4 Vessel Fleet for O&M

In the execution of O&M activities at offshore wind farms, the choice of vessel fleet size and mix can have great consequences for the O&M costs, as different vessel types have different capabilities and characteristics. Three factors that have a large influence on the utilization of O&M vessels are: distance from harbour, water depth and weather conditions. Distance to the wind farm from the harbour, together with the vessels transit speed, determines the transit time to get technicians out to the turbine. For maintenance tasks requiring lifting, water depths limit the choice of vessels that can be used. Weather conditions, like wave heights, wind speeds and water currents, place restrictions on vessel operation as vessels cannot operate safely in all weather conditions. Due to high downtime costs associated with unexpected failures, having vessels with high transit speed and that can handle rough weather, can contribute to achieving high wind farm accessibility and reduce downtime significantly. As an example, in the North Sea, the average number of days per year with wave heights over 2 m is 255 days. A vessel with a wave limit of 2 m, would hence only be able to operate 61% of the year. Increasing the wave limit to 3 m would make the vessel able to operate 86% of the year [64].

Today, a wide range of vessel types are used to execute maintenance tasks at offshore wind farms. During the operational phase of a wind farm, the vessels are generally utilized to perform three broad types of tasks: transferring crew to the turbines, accommodate technicians offshore, and to perform heavier maintenance tasks which require lifting capabilities or transportation of larger spare parts. Different vessels have different properties like fuel consumption, transit speed, maximum tonnage, deck space and passenger capacities. Vessels also differ in their ability to operate under harsh weather conditions. Furthermore, the operation and acquiring costs differ greatly between various vessel types. The most common vessel types utilized in the industry are presented in the Subsections 2.4.1 - 2.4.3. Subsection 2.4.4 describes how the vessel types can be acquired.

2.4.1 Crew Transfer Vessels

A crew transfer vessel (CTV) is a relatively small vessel used to transport technicians, tools, and spare parts to conduct tasks like inspections, minor repairs and technical problems that can be solved without heavy equipment [12]. A typical CTV has a load capacity between 1-50 tons, can travel at speeds between 15-30 knots, and are relatively inexpensive [139]. The four most common types of CTVs are inflatable boats (RIBs), mono hulls, catamarans and Small Waterplane Area Twin Hull (SWATCH) ships. RIBs are mainly used as a quick response vessel to provide fast access to site. They are highly available in the market and highly fuel efficient. However, they are unsuitable for long transits, have little transportation capacity, and are limited to wave heights of 0.75-1.25 m [92]. Mono hulls are stable, can handle severe weather conditions, and generally have a larger load capacity than other CTV types. [27].

(a) Mono hull, retrieved from [75]

(b) SWATCH, retrieved from [99]

(c) Catamaran, retrieved from [101]

(d) RIB, retrieved from [136].

Figure 2.5: Examples of different types of CTVs.

In later years, catamarans have gained increasing popularity in the industry, due to speed advantages and good seafaring capabilities [27]. In comparison with a mono hull, a catamaran cannot carry significant cargo. Due to personnel facilities and comfort, catamarans are usually unsuitable for journeys longer than 2 hours, and are restricted by wave heights above 0.60 - 1.75 m [92]. SWATCH ships are relatively new, and are currently entering the market. These ships are similar to catamarans, but due to design specifics these ships have significant stability advantages in rougher weather conditions [27]. This makes them able to operate in wave heights between 1.00 - 2.00 m [92]. However, SWATCH vessels have limited cargo capabilities, designs are more expensive to build, and they have higher maintenance requirements than other simpler vessels [27]. Examples of CTVs are shown in Figure 2.5 (a) - (d).

2.4.2 Helicopters and Crane Vessels

Similarly to CTVs, helicopters are used to transfer technicians to turbines for inspections and minor maintenance tasks. Helicopters have the advantage of shorter transit times, and are not restricted by wave heights. However, they are restricted by weather in terms of wind speed and visibility. Compared to seafaring vessels, they are generally more expensive to acquire and operate, and have less transportation capacity. In addition, helicopters cannot stay at the wind farm while maintenance is being performed and hence have to return to depot between delivery and pickup of technicians. An example of a helicopter is shown in Figure 2.6 (a).

Occasionally, heavier maintenance tasks, like replacing a blade or generator, need to be conducted. These types of maintenance tasks require crane vessels. Crane vessels are complex and highly specialized ships, which are expensive to acquire and operate. They are therefore rarely used if heavy-lifting is not required. Crane vessels vary in type and size. One type of crane vessel is the jack-up barge, which is a self-elevating mobile platform. An example of a jack-up barge is shown in Figure 2.6 (b). The water depth at the wind farm often places restrictions on the use of crane vessels. Furthermore, crane vessels also have limitations related to lifting height and weight.

(a) Helicopter, retrieved from [102].

(b) Jack-up barge, retrieved from [104].

Figure 2.6: Examples of a helicopter and a crane vessel.

2.4.3 Accommodation Vessels and Offshore Stations

As offshore projects are growing in size and moving further offshore, the need for offshore accommodation has grown. CTVs can only stay offshore for a limited period of time, and long distances from port to the working area makes transit times long and the accessibility poor. To meet this challenge, new offshore accommodation concepts have been developed. These concepts are expensive to acquire and operate. However, offshore accommodation is advantageous as it provides the opportunity of exploiting shorter windows of good weather, in addition to low travel times and closeness to the wind farm. For this reason, future projects with a distance from shore exceeding 55 - 75 km, are expected to become reliant upon offshore accommodation to avoid excessive travel times and low productivity due to seasickness [92].

Offshore accommodation concepts can take two basic forms: Floating Accommodation Vessels (AVs) and fixed offshore accommodation stations. AVs include a variety of different vessel concepts. AVs called floatels, has existed in the industry from 2008 [59]. Floatels are usually converted ferries or small cruise ships, which can stay offshore for long periods. Another AV type called motherships, are custom-made for the industry and have most of the equipment required for O&M services. Motherships often have the capability of accommodating helicopters and smaller CTVs, and are made to operate in rough weather conditions. Examples of AVs are shown in Figure 2.7 (a) and (b). Different types of fixed offshore accommodation stations are in development. These stations reduce transfer times, but do not improve wave height capabilities which is the case with AVs. Current industry trends therefore suggests that the market is moving more towards floating accommodation rather than towards the fixed offshore accommodation stations [92].

(a) Floatel, retrieved from [59].

(b) Mothership, retrieved from [103].

(c) Offshore accommodation station, retrieved from [16].

Figure 2.7: Examples of AVs and offshore accomodation stations.

2.4.4 Fleet Adjustment

Vessels used for conducting O&M can be acquired in different ways, and are normally long-term contracted, chartered from the open marked on a short-term basis, or bought from a ship-builder or from a second-hand market.

CTVs are most often contracted on a long-term lease from specialist marine contractors, but are sometimes bought instead. Larger vessel types, like crane vessels, are generally contracted on a short- or long-term basis. If a crane vessel is bought, it is primarily done for providing installation capacity rather than as a resource for O&M. However, it is predicted that as offshore projects increase in size, the economic rationale of having jack-up barges permanently on site will increase [109]. Motherships and offshore platforms are very new concepts, and contracting regimes for these types of vessels have therefore not emerged yet. Helicopters are mainly sub-contracted through a specialist helicopter operator, and it is unlikely that a wind farm owner would choose to buy one [109]. If a vessel is ordered from a ship builder, there is often a long lead time from the order is placed until the vessel is available to the owner. For chartering contracts, long lead times can also be the case for complex and specialized offshore vessels like motherships and crane vessels [91]. Most CTVs are highly available in the market and have short lead times independent of the way they are acquired.

There are several advantages and disadvantages related to different ways of acquiring a vessel. Dalgic et al. [39] compares three different chartering strategies for offshore wind farms: spot market, short-term charter and long-term charter. In their study, they do not take into consideration buying vessels, but this can be considered as a long-term charter of 20 years, as the lifetime of a vessel is approximately 20 years. Their results are shown in Table 2.1.

Table 2.1: Comparison of vessel chartering strategies, retrieved from [39].

Strategy	Advantages	Disadvantages
Spot market Min: 1 month Max: 3 months	- Use vessel only after a failure of wind turbine occurs - Select optimal vessel for each turbine failure - Only use vessel when required - Maximum utilisation of vessel	- Potential limited certainty in vessel availability - High uncertainty in mobilisation time and costs - Day rates and mobilisation costs are likely to be very high
Short-term charter Min: 3 months Max: 1 year	- Reduces risk of weather effect (if performed during summer) - Reduce number of vessels being chartered - Can be used across multiple sites	- Risk of low utilization in winter - In case of maintenance/supply delays, risk of uncompleted/ imperfect repairs
Long-term charter Min: 1 year Max: 20 years	- Reduced mobilisation time and costs - Eliminated risk of vessel unavailability - Increased operational control for the offshore wind farm operator - Cost vary less over lifetime - Can use across multiple sites - Better planning - Stable costs	- Paying for vessel even when not being used - High initial investment - Vessel not optimised for individual sites - A management team is required to operate the vessel - Repair and maintenance expenses may be added - A port is needed

2.5 Technological Developments

The introduction of new technology in the industry provides great potential for cost reduction in O&M. Development and refinement of already existing technology, in addition to the innovation and creation of new technology, affect both wind turbines and vessels used for O&M. This section presents the most important technological developments in turbine technology and vessel technology, and considers how these developments can contribute to reducing the O&M costs.

2.5.1 Turbine Technology

Offshore turbine technology is relatively new. In accordance with the technology life cycle, one can hence consider turbine technology to be in the ascent phase, and that the technology will reach the maturity phase in the future. In general, studies show that more mature turbine technology, which has been tested and developed over a longer period of time, have lower failure rates than new, more immature, technology [119]. It can hence be expected that as offshore turbine technology develops, reaching the maturity phase, the need for corrective maintenance will decrease, leading to great cost reductions. In addition, a more predictable demand for corrective maintenance might reduce O&M costs further, as the need for vessels also becomes more predictable, making it easier to plan the acquisition and utilization of vessels.

As the wind farms move further away from shore, where the weather is more beneficial for power production, there has been a trend towards developing bigger turbines, as discussed in Section 2.2. Statistical data show that larger (less mature) turbines have a higher failure rate compared to smaller (more mature) turbines [119]. This can also be seen in light of the technology life cycle, where the bigger turbines are in an earlier phase, still requiring testing and further improvement before reaching a maturity phase where the failure rate starts to decrease.

Wind turbine failures are related to large downtime costs, and continuous monitoring of the wind turbines' condition can play an important role in minimizing overall costs of O&M [6]. Condition monitoring of turbines can be conducted manually by inspection or automatically through a condition monitoring system (CMS). A CMS carry out different measurements to provide the operator with data on the condition of various system components [118]. As wind farms often experience low accessibility due to weather conditions, technological developments that allow a wireless CMS to be used to monitor the turbines, could reduce the cost of O&M [100]. Currently there are high costs related to operating certain types of CMS, causing the monitoring costs to outweigh the benefits from reduction in O&M costs [100]. If the cost of CMS technology is reduced in the future, it could be worthwhile to invest more in monitoring technology to better predict and plan maintenance.

2.5.2 Vessel Technology and New Concepts

Improvement and availability of new technology affect the vessel design and building choices that ship builders make. According to Marine Insight [83], technologies that might affect the vessel industry in the future includes concepts like: shipbuilding robotics, 3D-printing technology and LNG fuelled engines. In a market outlook for the shipping industry, DNV points at the offshore segment being the leading in employing new and innovative technology [44]. This is due to the high demand for specialized vessels in this segment.

One of the highest variable cost drivers for operating a vessel is fuel consumption. Technology and vessel concepts that can contribute to reducing fuel costs are therefore important to lower the overall operating costs of vessels. Such vessel technology includes gas fuelled engines and hybrid propulsion systems. According to a market outlook from DNV, it is expected that more than 1 out of 10 new-buildings until 2020 will be delivered with gas fuelled engines. For hybrid propulsion systems, pilot projects for offshore vessels indicate 15% fuel savings. According to DNV [131], hybrid propulsion systems are most beneficial for vessels with large variations in power demand. Larger and specialized O&M vessels fit into this category due to their high variation in power demand when maintenance tasks are being performed.

Newer vessel concepts used for O&M are often specialized and custom made. If the technology that is used today improves in terms of production costs and efficiency, this can allow current vessel concepts to be sold at a cheaper price. Furthermore, development of new technology can result in vessel concepts with capabilities not available today. As an example, new technology could allow building vessels that can handle rougher weather, and hence increase the accessibility of wind farms. Siemens recently released the first purpose built Service Operation Vessel for the industry, which is a vessel that can handle turbine access in wave height up to 2.5 m [110]. Another new concept is the Surface Effect Ship, which uses air-cushions to stabilize wave motion. This makes the vessel able to dock with offshore turbines in higher waves than what is possible today [133].

2.6 Costs of O&M

The cost of conducting maintenance at offshore wind farms can be divided into several cost components, such as labour costs, material costs, vessels cost and revenue losses due to production stops [39]. However, looking at the size of the cost components, the cost of maintenance can mainly be seen as a trade-off between the cost of acquiring and operating a fleet and the downtime cost. A complex and large fleet is significantly more expensive to acquire and operate than a smaller fleet. However, such a fleet increases the accessibility and availability of the wind farm, leading to a reduction in downtime cost. As Figure 2.8 shows, an appropriate availability level of the wind farm must be found, where the total maintenance cost is minimized through finding an optimal balance between the lost revenue of production stops and the direct cost of O&M. These two major cost components are discussed in the following subsections.

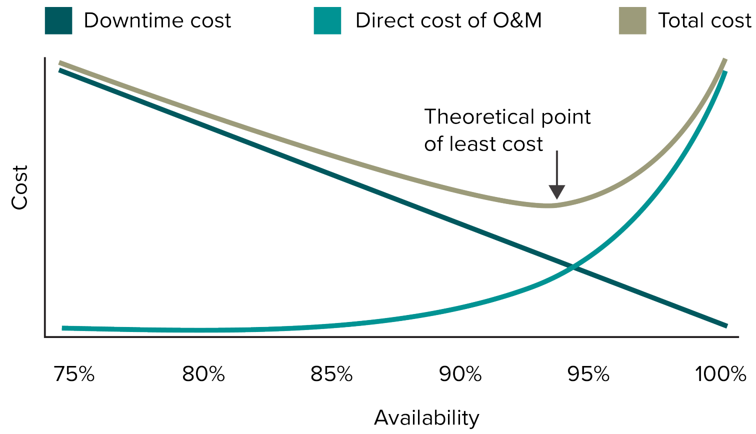


Figure 2.8: Relation between direct cost of O&M and downtime cost, based on [109].

2.6.1 Downtime Costs

The downtime of a wind turbine is given by the amount of time the turbine has to be shut down due to technical failures. The downtime costs are related to the loss of revenue from power that could have been generated during the downtime periods. The downtime cost can be described in terms of power output, electricity spot price, governmental subsidies and total downtime.

A turbine attempts to capture as much of the wind's power as possible and efficiently convert it into electricity, while also protecting itself from damage [90]. The power output of a turbine hence depend on the wind speed in the area and the technical capabilities of the turbine. Depending on the turbine, production of electricity is related to a defined cut-in, rated and cut-out wind speed as depicted in Figure 2.9. The turbine starts generating power at the cut-in wind speed, and from that point increase the power output corresponding to the increased wind speed until it reaches the rated speed where the power output stabilizes. At the cut-out wind speed, the turbine shuts down to avoid damages [90].The technical capabilities, such as the turbine capacity, hence determine how efficiently the wind speed can be utilized. The greater the capacity of the turbine, the more power is generated in a period of time. Hence, the larger wind speed and capacity, the larger the downtime cost is.

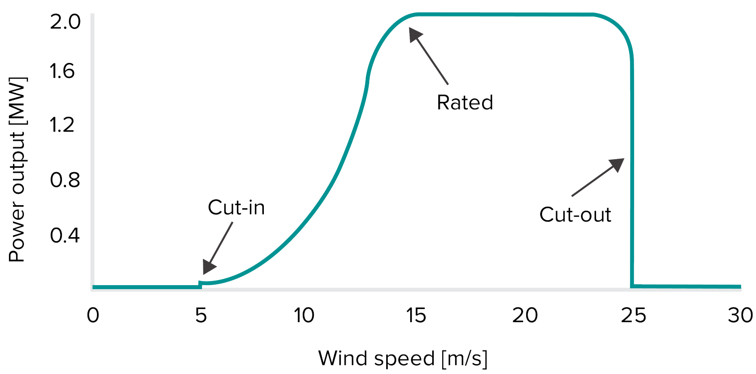


Figure 2.9: Power output for a 2 MW turbine, based on [90]

Depending on the country, different schemes for governmental subsidies are available for producers of electricity from renewable sources. The financial support is often given in terms of a subsidy per kW of capacity installed, or a payment per kWh produced and sold [82]. Two major strategies for subsidising offshore wind are outlined by EWEA: an investment-focused strategy and a generation-based strategy [82]. In an investment-focused strategy, financial support is given through investment subsidies, soft loans and tax credits. This is usually measured per unit of installed capacity. In a generation-based strategy, the financial support is either a fixed premium in addition to the electricity price, or a fixed regulated feed-in tariff. The design of the subsidy scheme used is important to the investors, as it affects their risk profile and cost of capital [112]. In the case where a generation-based strategy with a fixed premium is used, the total revenue per kWh is the spot price plus the subsidy. In this case, the subsidies hence affect the downtime cost directly.

The electricity spot price contributes to the downtime cost, and the spot price at the time of failure can therefore influence the decision of whether to prioritize to fix the turbine quickly or not. Fluctuations in electricity spot prices are large, and the price is influenced by several factors like time of the day, season and availability of different ways of generating electricity [31]. The future outlook for energy prices is uncertain. Trends in the energy sector point towards an increasing installment and use of unstable renewable energy sources, with lower marginal cost of production than non-renewable energy resources. Whether this will contribute to a generally higher or lower spot price in a long-term perspective is highly uncertain. However, as an increasing amount of renewable energy sources with unstable supply are connected to the grid, the volatility of spot prices are expected to increase. This makes it even harder to predict short-term spot prices in the future.

Given turbine power output, electricity spot prices, and subsidy schemes as described above, the downtime cost can be summarized in the following equation:

$$\begin{aligned} &\text{Downtime Costs } [€] = \\ &\text{Generated Effect [MW]} * (\text{Spot Price} + \text{Subsidy})[€/\text{kWh}] * \text{Downtime [hours]} \end{aligned} \tag{2.1}$$

where the generated effect is given as:

$$\text{Generated Effect [MW]} = \frac{1}{2} * \rho * A * v^3 * C_p \tag{2.2}$$

where ρ is the air density $[kg/m^3]$, A is the swept area of the turbine blades $[m^2]$, v is the wind speed $[m/s]$, and C_p is the average power coefficient [-].

2.6.2 Direct Costs of O&M

The direct cost of O&M mainly consists of the acquisition and operation costs of the vessel fleet that is utilized to conduct maintenance. Studies show that the cost of vessels make up 73% of total O&M costs today [39]. Reducing the cost of the vessel fleet can therefore have a significant affect on overall O&M cost. The cost of owning and operating a vessel fleet consists of fixed and variable costs. Among the fixed costs, the main costs are related to acquiring the vessel. The variable costs of a vessel fleet are related to fuel costs, labor costs, and costs of maintenance of the vessels.

Factors that influence the cost of acquiring vessels are: vessel type, vessel availability in the market and the way the vessel is acquired. Charter rates are closely related to the length of the contracting period, and chartering a vessel for a longer period of time can decrease daily rates significantly, compared to chartering for a shorter period. This is illustrated in Figure 2.10, where it is shown how the daily charter rate of a jack-up barge vary for different lengths of contracting periods. The figure also illustrates that a vessel's capabilities to perform O&M tasks influence the charter price. Seasonality is another factor that affects charter rates. As maintenance operations often are restricted by rough weather conditions during winter, the charter rate for vessels decrease along with the decreasing demand in this period. In summer, on the other hand, wind speeds are lower and weather places less restrictions on performing maintenance. This leads to an increased demand for vessels, and hence increased charter prices.

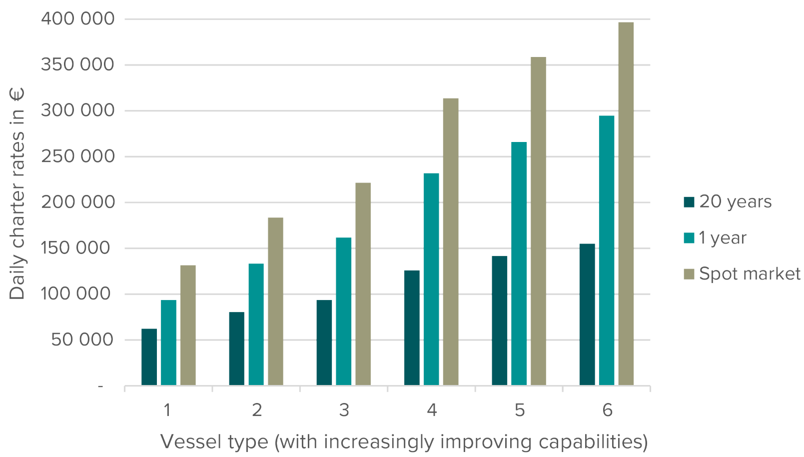


Figure 2.10: Daily charter rates for jack-up barges, based on [39].

As wind farms grow larger and are placed in closer proximity to one another, opportunities for cost reductions in O&M appear. In projects like the Dogger Bank mentioned in Section 2.2, where several large wind farms are placed close together, O&M costs can be saved through a joint fleet. A joint vessel fleet can lead to economies of scale through efficient utilization of the fleet by sharing vessels between several wind farm owners. Especially for complex and expensive ships, such as jack-up barges and AVs, sharing vessels between several wind farms can lead to significant cost reductions.

Chapter 3

Problem Description

In this thesis, the problem of determining an optimal fleet size and mix to execute maintenance tasks at one or several offshore wind farms is studied. The problem is studied from a strategic point of view with a planning horizon of 20-30 years, which coincides with the lifetime of a wind farm. The objective is to make optimal long-term strategic decisions that minimize costs. The main cost drivers of maintenance operations are related to the costs of acquiring and operating the vessels in the fleet, and the downtime costs incurred when turbines are shut down. The optimal fleet size and mix decision hence needs to made while finding an appropriate balance between these cost components.

The decisions that have to be considered when approaching the problem studied, can be divided into two types: long-term strategic decisions and short-term tactical decisions. The strategic decisions include the initial decision of the fleet size and mix at the beginning of a wind farm's lifetime, as well as how and when to adjust the fleet through long- and short-term chartering vessels over the course of the wind farm's lifetime. The decision-maker also needs to decide on the length of the charter contracts, the types of vessels to charter, and where to locate the acquired vessels. The tactical decisions include the deployment of the fleet to carry out maintenance tasks, and how to prioritize the different maintenance tasks given information about weather conditions and demand for maintenance. When solving the problem, the main objective is to find the here-and-now decision, which is the strategic decision regarding the initial fleet size and mix. The tactical decisions, as well as the future strategic decisions, are mainly relevant for supporting the here-and-now decision.

An offshore wind farm consist of a number of identical turbines, with the same capacity and size. A wind farm is located at a certain distance from shore, at a certain water depth, and in varying weather and wind conditions. The weather conditions and distance from shore affect traveling times and the accessibility to the wind farm(s). The water depth restricts the operation of certain types of vessels. A wind farm owner can have several wind farms placed in close proximity to each other, and can choose to conduct maintenance collectively at several wind farms. The development of wind farms is often conducted in stages, where a certain number of turbines are added to the wind farm in every stage.

The turbines at offshore wind farm(s) have a demand for maintenance that has to be met by the wind farm owner. There are two different types of maintenance activities that have to be executed at offshore wind farms: preventive maintenance and corrective maintenance. Preventive maintenance is scheduled according to a periodic maintenance strategy, and the demand is given by the number of turbines at the farm. The demand for preventive maintenance is hence known with certainty at the beginning of the planning period. Preventive tasks do not have to be conducted as scheduled, but if preventive tasks are executed too often or too seldom, the costs of maintenance increases. In addition, different types of unforeseen turbine failures might occur, which require different types of corrective maintenance. The demand for corrective maintenance cannot be known with certainty, but the probability of each type of failure can be forecasted. When unforeseen failures occur, corrective maintenance should be conducted as soon as possible to keep downtime cost low.

Different maintenance tasks have different requirements for resources, like spare parts, time, heavy-lifting and crew. These requirements place restrictions on what types of vessels that can be used to conduct different tasks, as the different vessel types can have considerably different capabilities. Due to the varying requirements of maintenance tasks, a vessel fleet used for maintenance usually consists of different vessel types. Different vessel types also have different capabilities of handling rough weather conditions, which places restrictions on the use of the vessels and accessibility of the wind farm. These restrictions might lead to the need of postponing certain maintenance tasks until a proper weather window occurs, reducing the availability of the wind farm. As the downtime cost increases with the time it takes before an unexpected failure is repaired, the vessel fleet's capability of handling rough weather has direct consequences on downtime costs. Hence, the more diverse and complex the fleet is, the more flexibility the wind farm owner has to perform maintenance tasks and keep downtime costs low. However, such a fleet is expensive to acquire and operate. A balance must therefore be struck between the cost of the fleet and the availability of the wind farm(s).

The decision regarding which vessel types to acquire is done based on the capabilities of the vessels and vessel charter prices. Different vessel types have different charter prices and operational costs, and vessel types that have more advanced capabilities (e.g. operate in higher wave heights, or transport more volume) are more expensive to acquire and operate. The choice of how and when to acquire the vessels affects the costs. The yearly charter cost decreases with the total length of the charter period and the availability of vessels, and the possible charter lengths may vary between vessel types.

The decision regarding fleet size and mix is subject to a wide range of uncertainties that affect the composition and size of the optimal fleet. On a strategic level, when seen over many years, a wide range of uncertainties need to be considered: electricity prices, technological developments, long-term charter rates and availability of vessels, the timing of when new wind farms are built, and the size of the increments built. The potential availability of new vessel concepts in the future introduces a trade-off between long-term chartering today's technology at a lower price, versus short-term chartering vessels today to be able to acquire prospective new technology in a few years. On a tactical level, the uncertain parameters include: electricity prices, weather conditions, charter rates of vessels, and demand for corrective maintenance.

Chapter 4

Literature Review

This chapter presents a review of relevant literature for the problem and solution methods studied in this master thesis. As mentioned, the problem studied in this thesis deals with the problem of deciding an optimal fleet size and mix for conducting maintenance at offshore wind farms, a logistic problem within the field of O&M. For this reason, previous work on logistic challenges related to O&M in offshore wind is reviewed in Section 4.1.

The specific problem studied in this thesis is commonly referred to as a maritime fleet size and mix problem (MFSMP) in the literature. MFSMPs appear in many industries and comprise decisions on how many ships of each type to acquire in order to meet demand of some type of service [105]. A vast amount of research have been conducted on these problems. However, only a few MFSMPs for the offshore wind industry have been studied previously in the literature. In order to get a thorough overview of MFSMPs, a review of MFSMPs for other industries is presented in Section 4.2.

A common classification scheme widely applied to studies of logistics management within operations research (OR), is to classify problems as strategic, tactical or operational, according to the length of the planning horizon in the problem [123]. A MFSMP can be classified either as a strategic or a tactical problem [105]. Strategic problems deal with long-term decisions that sets the stage for tactical and operational decisions [29]. Within offshore wind, these types of decisions often affect the wind farm for its entire lifetime [123]. Tactical problems consider decisions which are updated more frequently [29]. This classification is used throughout this thesis.

When considering MFSMPs with a long planning horizon, tactical decisions regarding routing and scheduling often have to be considered in order to evaluate the quality of strategic decisions [105]. Problems which consider decisions at more than one decision level are often referred to as dual-level planning problems in the literature. As the problem of this thesis consider decisions made on both strategic and tactical level, and hence can be seen as a dual-level problem, some research on such problems is reviewed in Section 4.3.

In addition to developing a model for the strategic MFSMP for O&M in offshore wind, an objective of this thesis is to develop an appropriate solution method. MFSMPs have been approached by a variety of solution methods in different papers, and some of these are reviewed as problems are presented throughout the chapter. For a summarised overview of solution methods for MFSMPs, the reader is referred to [105]. In Section 4.4, some applications of heuristic solution methods for complex combinatorial optimization problems are reviewed, with a focus on the metaheuristic used in this thesis.

4.1 Maintenance Logistic Problems in Offshore Wind

Several studies have been conducted for logistic problems within the field of O&M at offshore wind farms. A recent contribution by Shafiee et al. [123] presents a literature survey of maintenance logistics in the offshore wind industry per 2015. A total of 137 academic works are reviewed, and it is concluded that strategic maintenance logistic problems have received most attention. Furthermore, the review shows that OR have been applied to a vast amount of different maintenance logistic problems in offshore wind, like spare parts management, maintenance strategy selection, transportation strategy selection, and maintenance task scheduling. However, only a few of these are considered to be within the scope of the problem studied in this thesis. For this reason, the following section mainly focuses on previous work related to maritime transportation problems within the field of O&M. This includes the tactical problems of routing and scheduling maintenance vessels, and the strategic problem of how to acquire maintenance vessels (MFSMP). However, to show some examples of the applicability of OR within the field of offshore wind, some papers considering other types of maintenance logistic problems are also reviewed. For a more thorough review and description of the mentioned papers, the reader is referred to [123].

4.1.1 Maritime Transportation Problems

Within the field of offshore wind, only a few models for the problem of acquiring a fleet for conducting maintenance have been developed. Gundegjerde and Halvorsen [63] present a deterministic tactical MFSMP for offshore wind with a planning horizon of one year. They study the decision of how many vessels to acquire, as well as whether to use offshore stations or not, in order to meet a given maintenance schedule. The model does, however, not consider the logistics of spare parts or the day-to-day utilisation of the fleet [63]. In [64], Gundegjerde and Halvorsen extend this deterministic model, to a three-stage stochastic model. The stochastic model accounts for uncertainty in vessel spot rates, weather conditions, electricity prices and wind turbine failure rates. The model is solved with an exact method, and their computational study shows that the stochastic model performs significantly better than the deterministic model.

In a thesis by Vefsnmo [134], a strategic MFSMP for offshore wind is studied with a time horizon of 25 years. The thesis presents a stochastic model for determining the optimal number of vessels to be acquired, taking short-term uncertainty in turbine failures and weather conditions into consideration. Furthermore, the model developed by Vefsmo accounts for stepwise development of wind farms. A computational experiment shows that the stepwise development of wind farms influence the optimal fleet, and hence is beneficial to take into consideration. Uncertainty related to the realization of the planned stepwise development of is not considered.

A few papers have been written about the tactical decisions of maritime routing and scheduling of maintenance vessels in offshore wind. Skaar presents a model for optimization of routing and scheduling when performing maintenance at offshore wind farms [128]. The model allows maintenance technicians to be left at the wind turbines and be picked up at a later time, giving a problem structure similar to a pickup-and-delivery problem. Raknes et al. [116] study the tactical problem of routing and scheduling a given joint vessel fleet to perform maintenance at multiple offshore wind farms. A deterministic mixed-integer programming model (MIP) for finding optimal maintenance schedules and vessel routes is presented. Furthermore, a rolling horizon heuristic is developed, which solves the problem iteratively by LP-relaxing some parts of the planning period, and fixing the solution for some of the variables that are not LP-relaxed. Their computational study shows that the heuristic outperforms the exact solution method, both in terms of solution time and solution quality. Dai et al. [38] also present a model for a routing and scheduling problem for maintenance at offshore wind farms. Their model determines the cost optimal assignments of turbines and routes to vessels.

4.1.2 Other Maintenance Logistic Problems

Some examples of strategic maintenance logistic problems in offshore wind studied in the literature are: location and capacity of maintenance accommodations, selection of wind farm maintenance strategy and outsourcing of maintenance services [123]. Rademakers et al. [114] study the strategic problem of selecting a wind farm maintenance strategy. They present two models for strategic O&M aspects of offshore wind farms. In their paper, an optimized O&M strategy is presented, giving when and how to conduct maintenance. Spare part availability, extreme weather conditions and crane availability is taken into consideration. The models are used to find points of improving baseline, determine the most cost effective weather windows and quantifying the uncertainties in downtime and costs.

Another approach to strategic decision making in O&M is the Cost Estimator (OMCE) developed at the Energy Research Centre of the Netherlands [115]. The OMCE-calculator aims to help owners and decision-makers to better estimate and control the future O&M costs. In a paper by van de Pieterman et al. [113], an investigation on the OMCE-calculator's potential for optimizing the maintenance strategy of a wind farm is presented, and the OMCE calculator is used to determine the optimal number of access vessels needed. The study concludes that long waiting times for suitable weather windows have a major influence on the downtime of the wind farm, leading to large revenue losses. Furthermore, the study finds that equipment cost and revenue losses contribute significantly to the total O&M costs.

Besnard et al. [14] present an optimization model that aims to support offshore wind farm owners on strategic decisions, such as the location of maintenance accommodation, the number of technicians, the choice of transfer vessel, and the use of a helicopter when conducting O&M. Jin et al. [77] study the problem of outsourcing maintenance services. They propose a mathematical model for minimizing O&M costs of wind turbines under a performance-based service contract. In such a contract, a service provider commits to ensure that the wind turbines meet a given availability goal.

Within the field of O&M at offshore wind farms, examples of tactical decisions are: the allocation of resources (e.g. vessels, technicians and crew) to maintenance activities, scheduling of maintenance tasks, and spare parts inventory management. Lindqvist et al. study a spare parts inventory management problem for wind farms in [86]. In their master thesis, an optimization model is utilized to determine optimal stock levels and reorder size for critical components.

4.2 Maritime Fleet Size and Mix Problems

The problem of determining the size and composition of a fleet is commonly referred to as the fleet size and mix problem (FSMP). For land-based industry, a vast amount of models have been proposed in published works. Hoff et al. give an extensive literature review of these in [72]. However, land-based FSMPs are not directly applicable to maritime industries, partly because capital costs are higher. In addition, MFSMPs are subject to a higher degree of uncertainty than FSMPs, as the lifetime of ships are generally longer than the lifetime of trucks [9]. Therefore, only FSMPs developed specifically for maritime industries are reviewed in this literature study. A range of MFSMPs have been studied for different maritime industries. As the research reviewed in this section consider MFSMPs in other industries than O&M in offshore wind, their relevance is limited. For a more thorough overview of literature related to MFSMPs, the reader is referred to a literature survey presented by Pantuso et al. in [105].

As mentioned, when considering strategic MFSMPs with a long planning horizon, tactical decisions regarding routing and scheduling often have to be considered in order to evaluate the quality of long-term decisions. However, Hoff et al. [72] state that in strategic MFSMPs, it typically does not make sense to include routing and scheduling aspects at a very detailed level. Hence, pure routing and scheduling problems for other industries than offshore wind has not been reviewed in this literature review. For interested readers, ship routing and scheduling problems are reviewed by Christiansen et al. [30] and Ronen [120].

4.2.1 Deterministic MFSMPs

The first publication on MFSMPs by Dantzig and Fulkerson [41] uses a linear programming (LP) model to minimize the number of identical navy fuel oil tankers needed to meet a fixed schedule of transportation. The problem has later been modified by Bellmore et al. [13] to include a utility for each delivery, introducing the opportunity to cancel deliveries by giving up the corresponding utility. Within the category of strategic MFSMPs, the first contribution was the problem of composing the US merchant marine fleet, presented by Everett et al. in 1972 [45]. A LP model is presented with the aim of determining the best ship designs and sizes to purchase, in order to carry 15% of US foreign trade. Another strategic MFSMP, for a Chinese coal shipping system, is presented by Zeng and Yang [137], and seeks to optimize the fleet size and mix, as well as ship schedules. The fleet design and ship routing decisions are solved simultaneously in an integrated model by a two-phase tabu search algorithm. In the first phase, a tabu search is conducted to find good ship designs. For each ship design, a second tabu search is conducted to determine the optimal ship routing and objective value.

In a paper by Sigurd et al. [127], the design of a sea-transport system between Norway and Central Europe is formulated as a set partitioning model and solved by a heuristic branch-and-price algorithm. The problem is decomposed into a master problem and a pricing problem, where the master problem selects transportation routes with minimized cost from a subset of feasible routes. The pricing problem is solved using delayed column generation, generating new and improving routes until no improving routes can be found. Murotsu et al. [97] study the strategic MFSMP for crude oil carriers. They combine dynamic programming and non-linear programming as a solution method, and vessel characteristics like size and speed are included as variables in the problem.

The earliest paper on tactical MFSPs, published by Schwartz in 1969 [122], presents an integer programming (IP) model for the MFSMP of a barge line company. The objective is to determine the number of barges and towboats of different sizes to be chartered in and out in order to provide a service at minimum cost. Lai and Lo [84] consider the optimal fleet size, routing and scheduling in a ferry network design problem. Their MIP problem is solved by developing a two-phase heuristic algorithm, where the first phase determines the set of feasible fleet size and routing paths, and the second phase evaluates combinations of paths to search for solution improvements. Performance testing indicates that the heuristic algorithm produces solutions within 1.3% of optimality.

Fagerholt and Lindstad [48] study the tactical MFSMP of determining which vessels to use, and their optimal weekly schedules, to service offshore installations from an onshore depot. The problem is solved by first generating a number of candidate schedules for all vessels, and then solving an IP model. Only small instances are solved, and hence all feasible schedules for the vessels are generated, making the solution method exact. A similar problem is solved with a similar solution approach by Halvorsen-Weare et al. in [67]. In their problem, aspects like spreads of departures and maximum and minimum durations of voyages are included. A voyage-based solution method consisting of two phases is suggested. A number of candidate voyages are generated in the first phase, by defining all possible subsets of offshore installations and solving a traveling salesman problem with multiple time windows for each subset. In phase two, the voyage-based model is solved by choosing the most cost-effective supply vessels and the best pre-generated voyages.

4.2.2 Stochastic MFSMPs

Only a limited number of publications on FSMPs consider uncertainty [135]. Hoff et. al [72] state that one of the major critiques of today's research on FMPS is the lack of treatment of stochastic aspects, together with concepts of risk and robustness. The amount of publications within MFSMPs accounting for uncertainty is also scarce. Among 37 publications surveyed by Pantuso et al. in [105], 27 consider planning in a deterministic context. The remaining 10 papers consider stochastic parameters, but the majority of these papers do this by replacing data with averages or extreme values in otherwise deterministic models. When considering maritime transportation, rather few real world problems can justifiably be considered deterministic. This is especially true for strategic planning problems with a long planning horizon [105]. This calls for further research on the use of stochastic approaches to MFSMPs.

One of the most commonly used OR approaches for dealing with uncertainty is the use of Stochastic Programming (SP). SP traces its roots back to 1955 when Dantzig introduced the recourse model in [40]. In general, stochastic programs are generalizations of deterministic mathematical programs, in which some data are not known with certainty. SPs are characterized by features like: decisions being made in discrete time periods, many decision variables having many potential values, having expected values in the objective, and dealing with (partially) known distributions [20]. A severe amount of research conducted on SPs contribute to the understanding of properties and the design of algorithmic approaches for solving SPs [69]. For a thorough presentation of SPs, the reader is referred to studies by Higle [69] and Birge and Louveaux [20].

Bakkehaug et al. use a SP approach to solve a strategic MFSMP for shipping companies [9]. They study the problem of how and when to adjust the fleet to efficiently meet transportation demand. A multi-stage SP model is proposed, where uncertainty in future parameters linked to the market situation in maritime shipping are considered. This includes parameters like: supply and demand of ships, vessel prices, demand of cargo and operating costs. In the model, all uncertain parameters are assumed perfectly correlated, and simplified into one random variable describing the market situation. The result from their study indicate that taking the inherent uncertainty of the problem into consideration, through the use of SP, significantly improve fleet adjustments compared to deterministic approaches. Pantuso et al. [107] also propose a SP model for the MFSMP and apply it on a case from a liner shipping company. They consider uncertainty in: building prices, secondhand prices, selling prices, charter rates, sunset values, scrapping values, variable operating costs, and demands. Similarly to Bakkehaug et al. [9], Pantuso et al. show that using a SP model gives tangible benefits compared to a deterministic model. Furthermore, their study show that the importance of uncertainty diminishes in markets with easier access to vessels and more standardized service frequencies.

In Meng and Wang [93], a tactical MFSMP for a container liner shipping company is modelled using a stochastic approach to tackle demand uncertainty. The problem is extended by Meng in [95] to include transshipment, and modelled as a two-stage stochastic integer programming model. Alvarez et al. [1] present a robust MIP model for the strategic MFSMP, with random variations in the selling and purchasing prices of ships. The model is made to assist companies in modifying and deploying the fleet. Fagerholt et al. [49] use a Monte Carlo simulation framework, built around an optimization-based decision support system for tactical routing and scheduling, to solve a strategic MFSMP for a Norwegian tramp shipping company. The model accounts for uncertainty at tactical level in the timing and quantity of cargoes. The optimization model uses a rolling horizon principle, revealing information as time goes by, making it possible to deal with stochastic aspects. Shysou et al. [125] present the problem of finding a cost optimal number of vessels to long-term charter for anchor handling operations, while considering uncertainty in weather conditions. The authors use a discrete event simulation framework to evaluate different fleet sizes. Halvorsen-Weare and Fagerholt [66] extend the deterministic model from [67], presented in Subsection 4.2.1, and use simulation to ensure more robust routes and fleet solutions with respect to uncertain weather conditions. Their results show that the inclusion of robustness criteria, accounting for unforeseen events, gives higher potential for improvements in costs, compared to the deterministic model.

4.3 Dual-Level Problems

When solving a strategic optimization problem, decisions on tactical or operational level often need to be considered in order to evaluate the quality of strategic decisions. This is necessary as strategic decisions place restrictions on lower level decisions, and the impact on lower level decisions hence must considered [80]. Accounting for both long-term (strategic) and short-term (lower level) decisions in one model is challenging, as these decisions often are made on different time scales: strategic decisions are often made with several years in between, while lower level decisions are made more frequently i.e. on a daily, weekly or monthly basis [80]. To complicate further, decisions at all levels may be subject to uncertainty. Models that include decisions at more than one decision level have many names in the literature, like hierarchical planning problems [106], multilevel planning problems [53] and dual-level problems [80]. Inspired by [80], the the term dual-level stochastic problem is used to refer to problems where uncertainty is accounted for at more than one decision level, in the remaining chapters of this master thesis.

Several deterministic dual-level models, and dual-level models that consider uncertainty at one decision level have been developed in the literature. In a paper by De Jonghe et al. [78], planning models to optimize generation investments in electric power systems are studied. The models presented in the paper integrate short-term responsiveness in a long-term model, by using an equilibrium model and a representative profile for a one-period deterministic model. Pesenti [108], studies the problem of how to purchase and deploy a number of container ships to meet a certain customer demand. The deterministic model is formulated as a three level hierarchical resource planning model, with strategic, tactical and operational decisions being made at each level, respectively. The model is solved with a heuristic solution method, where information is passed bottom-up and top-down between the three levels in order to test and receive feedback on the decisions at different levels. In [121], Schütz et al. present a dual-level problem for the Norwegian meat industry. The problem is modeled as a two-stage stochastic problem, where uncertainty in demand is considered at operational level. Sönmez et al. [129] study the strategic investment decisions concerning technology and capacity choices, in liquefied natural gas transport. The impact of using a stochastic simulation model for throughput at operational level is discussed, and it is shown that operational flexibility is important to cope with short-term variations and has significant impact on profitability when making strategic decisions.

Only a few examples of dual-level stochastic models which considers uncertainty at more than one decision level exist in the literature. Kaut et al. [80] argue that standard modelling approaches, like traditional scenario tree-based SP models, are inappropriate for dual-level stochastic problems. They argue that with such a modelling approach, it is not obvious how to combine the different time scales without the model growing extremely large in size. This issue has been addressed in different manners in the literature. Kaut et al. [80] present a model for infrastructure-planning in the energy sector, which includes strategic and operational decisions, both subject to uncertainty. They present a dual-level optimization approach where important operational features are embedded directly into the strategic decision model. This allows an immediate evaluation of potential strategic solutions. The modeling approach is based on a new scenario tree structure, a dual-level scenario tree, developed by the authors. The dual-level scenario tree separates between strategic nodes for long-term decisions and operational nodes for short-term decisions, leading to a separation between strategic and operational scenarios. Since strategic decisions typically do not depend directly on any particular operational scenario, but rather on the overall performance in all operational scenarios, it is considered sufficient to only branch between strategic stages, and the operational nodes are hence embedded into their respective strategic nodes. An application of the same scenario tree structure is presented by Hellemo et al. [68] for a problem in the natural gas industry. In both [68] and [80], the dual-level stochastic model is only solved for some small instances by the use of an exact solution method.

In Pantuso et al. [106], a similar modeling approach as in [80], named multistage hierarchical stochastic modelling, is developed for a dual-level stochastic MFSMP. The authors distinguish between two types of decisions in their problem: aggregate level decisions (ALDs), comprising the decision of which vessels to use and how to acquire them, and detailed level decisions (DLDs) of deciding how to utilize the fleet to service customer demand. Both ALDs and DLDs are subject to uncertainty. The authors show that if a certain structural property can be identified in a dual-level stochastic model, the resulting stochastic program has *block-separable recourse*, as described by Louveaux in [88]. The property of block-separable recourse is beneficial, as it allows treating the multistage dual-level model as a two-stage stochastic problem, where first-stage decisions represent ALDs in all strategic nodes and second-stage decisions represent DLDs. For the structural property to hold, it must be possible to distinguish a set of DLDs that has no influence on future decisions of any kind, and hence can be forgotten once made.

Furthermore, Pantuso et al. [106] present a heuristic solution method for their problem. The property of block-separable recourse is exploited to decompose the problem into a master problem (MP) and many independent LP subproblems. A subset of the subproblems are solved by an exact solution method, when necessary. The MP is solved by a tabu search heuristic. In the tabu search, the type and number of ships to be acquired or disposed of are adjusted dynamically in three phases. In the first phase, the fleet size and mix is adjusted in the root node. In the second phase, the fleet size and mix is adjusted in the rest of the tree. In the third phase, the size of the fleet is kept constant, while the mix is adjusted again in the root node. The heuristic solution method is tested against the deterministic equivalent problem (DEP) solved by a commercial optimization solver. The results indicate that the heuristic solution procedure is less efficient than the commercial solver for small instances, but significantly better for larger problem instances. Furthermore, the heuristic is able to generate solutions to test instances where the optimization solver reaches memory limitations.

4.4 Heuristic Solution Methods

Heuristic solution methods are widely applied to complex combinatorial optimization problems in the literature. The solution method developed for the problem studied in this thesis is based on the metaheuristic Greedy Randomized Adaptive Search Procedure (GRASP). This section therefore mainly focus on reviewing applications of GRASP in the literature. A few examples of other metaheuristcs are included to show the motivation behind applying metaheuristics as a solution method for stochastic optimization problems.

4.4.1 Metaheuristics for Stochastich Optimization Problems

Bianchi et al. [17] present a literature survey on metaheuristics used to solve stochastic combinatorial optimization problems. They point out that due to the high complexity of SP models, classical exact solution methods are often incapable of solving large instances of the problems. Furthermore, they argue that approaches based on metaheuristics often are capable of finding good, and sometimes optimal, solutions to instances of realistic size within less computational time [17].

Several papers illustrate the benefits of applying metaheuristics instead of exact methods to stochastic problems. In Gutjahr [65], a traveling salesman problem with stochastic time windows is solved with both exact and heuristic methods. The exact method, complete enumeration, solves instances with 10 customers in about 4 hours. In comparison, the metaheuristics applied, Ant Colonization and Simulated Annealing, solve instances with up to 20 customers in a few seconds. Gendrau et al. [58] compares the performance of the integer L-shaped method and tabu search (TS) on a vehicle routing problem with stochastic demands and customers, and conclude that TS is significantly faster. Costa et al. compare the performance of branch & bound (B&B) and TS for solving a sequential ordering problem with time constraints [37]. Their results show that the TS is much faster than B&B, and that while the B&B is only able to solve instances with up to 14 causes, the TS solves instances with up to 500 causes.

4.4.2 Greedy Randomized Adaptive Search Procedure

GRASP is a multi-start search procedure, in which each iteration consists of two phases: construction and local search [117]. The metaheuristic was first introduced by Feo and Resende in 1989 [50]. Several enhancements and hybridization with other methods like tabu search, simulated annealing and genetic algorithms have since been proposed. For an extensive overview of GRASP fundamentals, enhancements, hybridization and applications, the reader is referred to Festa et al. [52] and Resende et al. [117].

GRASP has been applied to find high quality solutions to a wide range of combinatorial optimization problems, such as: power systems[18], set covering [51], scheduling [5], manufacturing [11], optimization in graphs [8], location [42], assignment [89], transportation [7], and inventory-routing [74]. Some examples of GRASP applications are elaborated on here to illustrate its wide applicability and extensive use in the literature. To the authors knowledge, no published research has applied GRASP on MFSMPs. Hence, GRASP applications for other complex combinatorial optimization problems have been reviewed. Furthermore, only one application of GRASP on stochastic problems is found in the literature in [74], hence most of the reviewed examples study deterministic problems.

The first application of GRASP was on the set covering problem arising when computing the incidence matrix of Steiner triple systems [51]. In the construction phase of the developed GRASP, a cover is constructed by selecting sets that covers the largest number of yet uncovered elements. In the local search phase, the heuristic seeks to make the constructed cover smaller by exchanging k-tuples of sets from the current cover with a p-tuple ($p < k$) not yet in the cover. Computational results show that the GRASP quickly produces the best known solutions for all instances considered [51].

Argüello and Bard propose a GRASP to reconstruct aircraft routings in response to groundings and delays experienced during a day. The objective is to minimize cost of reassigning aircrafts to flights whenever a disruption in a schedule appear. In the GRASP, neighbours of an incumbent solution are generated and evaluated, and the most desirable neighbours are placed in a restricted candidate list. The results presented in the paper demonstrate the ability of the heuristic to quickly explore a wide range of options, and to produce optimal or near-optimal solutions [7].

Hvattum et al. [74] apply GRASP to a scenario tree based formulation of the stochastic inventory-routing problem (SIRP). Only the construction phase is included in the GRASP, while the local search phase has been left out. The solution is constructed iteratively, by increasing the amount of units to deliver to a customer in each iteration, starting from zero delivery. Two versions of GRASP are implemented and tested on the SIRP. In an any-node GRASP, insertions at all nodes in the tree are considered in each iteration, leading to a large number of possible insertions. In order to reduce the number of possible insertions, a top-down GRASP is developed. In the top-down GRASP, insertions are restricted to be done in a top-down fashion. All decisions in the root node are made first, before continuing the construction node by node, recursively in the tree. In a computational study, the performance of the GRASP heuristic is compared with an exact solution method and several simple matheuristics implemented directly in an optimization solver. The results show that the top-down GRASP performs best for the tested SIRP instances.

Prais and Ribeiro [111] apply GRASP to a matrix decomposition problem arising in the context of traffic assignment in communication satellites. Furthermore, they propose an extension of the basic GRASP, called Reactive GRASP, in which the basic parameter that defines the restrictiveness of the candidate list during the construction phase is self-adjusted according to the quality of the solutions previously found [111]. Their results show that the Reactive GRASP outperforms the basic GRASP algorithm for the problem studied. Moreover, their results show that the Reactive GRASP is more robust, and that calibration efforts are drastically reduced. The Reactive GRASP approach has later been used for several other problems. Some examples are: power systems transmission planning [18], job shop scheduling [19], single source capacitated plant location [42], strip-packing [2] , and a combined production-distribution problem [22].

4.5 Our Contribution

As presented in this chapter, several studies have been conducted for MFSMPs. However, only a few models have been developed for the MFSMP in offshore wind. Furthermore, only a few of these models consider uncertainty. Among the few existing stochastic models, only short-term tactical uncertainty has been accounted for. Studies on how to capture both the strategic and the tactical uncertainties in one optimization model do, to the authors knowledge, not exist for offshore wind.

This master thesis presents a new way to model the strategic fleet size and mix problem for maintenance at offshore wind farms, by using a dual-level stochastic modelling approach. The model accounts for uncertain parameters that has not yet been considered for this problem, like uncertainty related to: technological development, stepwise development of wind farms, long-term trends in electricity prices, subsidies, and long-term trends in vessel charter prices. Furthermore, a method for evaluating the added benefit of accounting for both long- and short-term uncertainty by using a dual-level optimization approach is developed.

Several heuristic solution methods for stochastic problems can be found in the literature, but to the authors knowledge the only heuristic developed for a dual-level stochastic model is the one presented by Pantuso in [106]. GRASP has, as mentioned, previously only be applied to one stochastic problem, and has not yet been applied to any MFSMPs. In this master thesis, a heuristic solution method based on GRASP is developed for the dual-level stochastic MFSMP for maintenance at offshore wind farms.

Chapter 5

Mathematical Model

This chapter presents the mathematical formulation of the Dual-Level Stochastic Fleet Size and Mix Problem for Maintenance at Offshore Wind Farms (DLPOW). The model was first developed in the authors specialisation project [23]. The assumptions that lie at the basis of the formulation is presented in Section 5.1. A detailed explanation of the indices, sets, parameters and decision variables used in the formulation is given in Section 5.2. The mathematical model is presented in detail in Section 5.3.

5.1 Assumptions

In this section, the underlying assumptions for the mathematical formulation are outlined. In the first part of this section, the problem structure and uncertain parameters are explained. In the second part, the assumptions related to charter agreements, vessels, maintenance tasks, wind farms and downtime cost are explained.

5.1.1 Problem Structure

As described in Chapter 3, the aim of this master thesis is to study the fleet size and mix problem in offshore wind from a strategic point of view, using decisions at tactical level to evaluate the quality of strategic decisions. Formulating a model for this problem is challenging due to combination of different time scales: while the strategic decisions of fleet size and mix have time horizons of many years, the tactical deployment decisions are made on a daily or weekly basis. In addition, decisions on both planning levels are subject to uncertainty. This introduces the issue of how to capture both time scales in one optimization model, while also accounting for uncertainty at both strategic and tactical level. As mentioned in Chapter 4, Kaut et al. [80] present an approach for modeling problems that combine decisions with different time-scales while accounting for uncertainty. Their approach is to use a dual-level optimization model where the tactical scenarios are embedded directly into the strategic decision model.

The structure of the mathematical model for DLPOW is based on the dual-level scenario tree presented by Kaut et al. in [80]. The general structure of the scenario tree used to model the DLPOW is depicted in Figure 5.1, where the strategic nodes represent points in time where strategic decisions are made. The strategic nodes are modelled with a set of embedded tactical scenarios for each season. The seasons have been introduced in order to capture seasonal differences in aspects like weather conditions, charter rates and electricity prices. Each tactical scenario contains a set of periods which denotes the time interval between two consecutive time-discretisation points. Tactical decisions are made in each period.

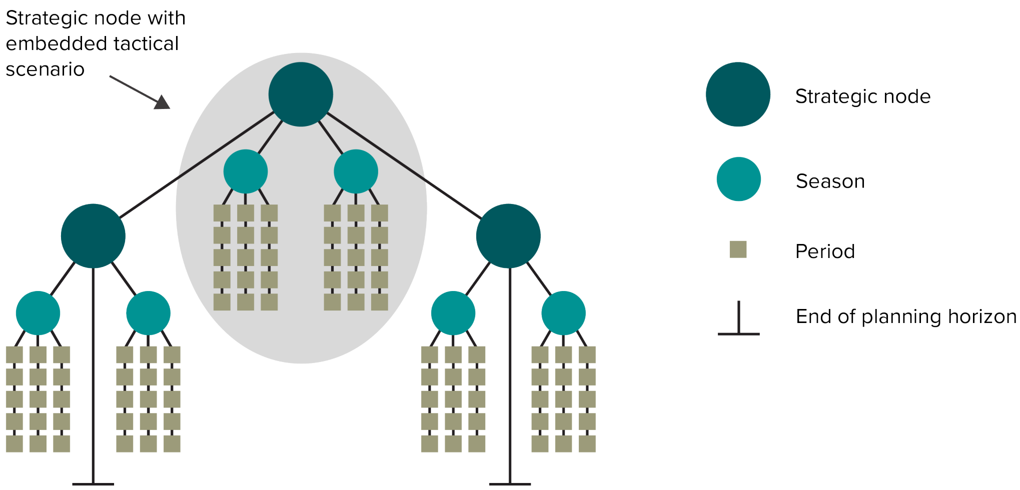


Figure 5.1: Problem structure of the DLPOW, inspired by [80].

The strategic decisions in the DLPOW are to decide how many vessels of each type to long-term charter, where to locate the vessels, and how long the chartering periods should be. For each season related to a strategic node, a decision on the amount of vessels to short-term charter in and out in each season must be made. These decisions are made with the same information as in the strategic nodes, and can hence be seen as strategic decisions. In the tactical scenarios, a recourse of the decision on fleet size and mix finds the optimal deployment cost of the fleet. In the mathematical formulation, the tactical problem of deployment is modelled on an aggregated level, and is primarily meant to support the strategic here-and-now decision.

As a consequence of the scenario tree structure used, an assumption is that the strategic uncertainty realized between the strategic nodes are independent of the tactical uncertainty realized in the tactical scenarios. This means that no strategic decision made in a strategic node is dependent on tactical decisions made in a tactical scenario. If the opposite was true, it would not be possible to model the problem with a single strategic node following two or more tactical scenarios. Additionally, the model assumes that the first tactical decision in a strategic node does not depend on the last tactical decision from the previous period in the proposed structure. Hence, there is no connection between tactical scenarios of two consecutive strategic nodes.

The model for DLPOW captures decisions being made on different time scales, and assumes that the time between strategic and tactical decisions are different. For the strategic decisions conducted in the strategic nodes, the time between each decision is given in years. The number of years between each node can vary depending on where you are in the planning horizon. This gives the modeller flexibility in the frequency of strategic decisions. All strategic nodes at the same stage in the tree represent a strategic scenario in the same year. The parameter $t(n)$ denotes the year of each node, and the number of years between two strategic nodes a and b is $t(a) - t(b)$. In the remaining text, this value is referred to as the duration of the strategic node a. All leaf nodes in the scenario tree are assumed to have the duration of one year. The tactical scenarios are, as mentioned, composed of a number of periods. Each period represents one day in the planning horizon, hence tactical decisions are made on a daily basis.

5.1.2 Uncertainty

For the strategic nodes, the uncertain parameters included in the model are: technological development of vessel concepts, stepwise development of wind farms, trends in electricity and vessel prices, and governmental subsidies. These strategic uncertainties are long-term, and new information is revealed with one to several years in between. The realized electricity price and vessel price can differ between the seasons of a strategic node. For the tactical scenarios, the short-term uncertain parameters are weather conditions, and the frequency and timing of turbine failures, giving the demand for corrective maintenance. These parameters differ from day to day, and are assumed to be known for all periods in a tactical scenario at the beginning of the respective tactical scenario. A timeline that depicts the decision structure in relation to new knowledge is shown in Figure 5.2.

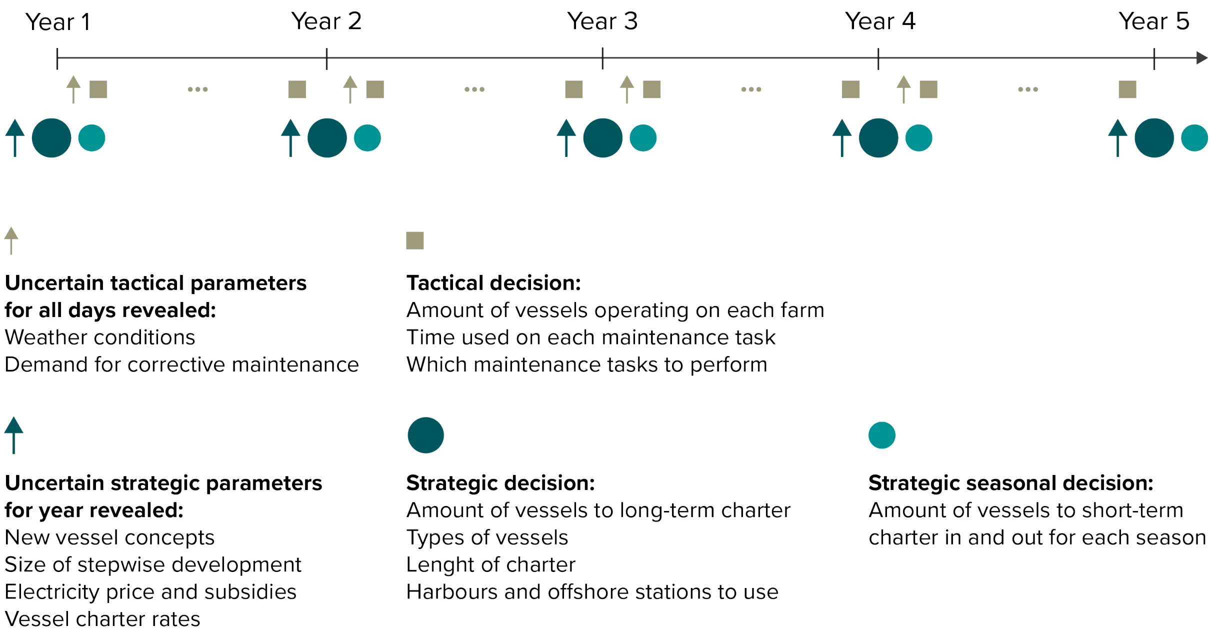


Figure 5.2: Timeline of decision structure in relation to new knowledge.

5.1.3 Charter Agreements

The here-and-now decision in the DLPOW is to decide an initial fleet at the beginning of a wind farms lifetime. For this reason, the mathematical model assumes that the decision-maker has zero vessels at the beginning of the planning horizon. When making decisions regarding long-term charter, the decision-maker needs to decide how many vessels of each type to charter, and how long the charter length for each vessel should be. The model assumes that the possible long-term charter lengths span from 1 to 25 years. Not all vessels can be chartered for all of the given charter lengths, and each vessel type v has a given set of possible lease expiration times, L_{nv}, when chartered in a strategic node n. The index $l \in L_{nv}$ gives an expiration time, which is the time $t(k)$ of the node(s) k in which the vessel of type v leaves the fleet if chartered in node n. It is assumed that the vessel leaves the fleet at the beginning of year $t(k)$. If a vessel is not available in the market in a node n, $L_{nv} = \emptyset$. In this way, the formulation models new vessel concepts appearing in different years and strategic scenarios.

Harbours and offshore stations do not have different lease lengths, as it is assumed that they can only be acquired for the whole planning horizon or not at all. This implies that if a harbour or offshore station is used at any point in time during the planning horizon, the total cost of acquiring and operating the station throughout the planning horizon is incurred. For short-term chartering, the model assumes that the only possible lease length is to charter for an entire season. The decisions that have to be made regarding short-term chartering are hence only how many vessels of each type to short-term charter in or out for a given season. The actual lease length is not a part of the decision. However, considering that not all vessel types are available in the market at all times, only vessel types for which $L_{vn} \neq \emptyset$ can be short-term chartered.

5.1.4 Vessel Types

In the mathematical formulation, a vessel type can be either a ship vessel type or a helicopter vessel type. The different vessel types are defined by the capabilities: crew capacity (number of people), capacity of transporting parts (tonnage), wave limit (m), wind limit (m/s), speed (knots), lifting capacity (tonnage), maximum operation time (man-hours/day), utilization ratio, occurrence in market (year) and origin (harbour or offshore station). Each vessel type belongs to a specific harbour or offshore station that has a given distance to each wind farm. This implies that a vessel that is stationed at at harbour or offshore station 1 is a different vessel type than a similar vessel with the same capabilities stationed at harbour or offshore station 2. For each vessel type, its respective capabilities determine which types of maintenance tasks it can perform, and how long and under which weather conditions the vessel can operate on a given day. The crew capacity, M_v^{CREW}, is used to calculate the amount of man-hours that a vessel type can perform maintenance at a wind farm. It is assumed that if a vessel type can transport M_v^{CREW} people, this amount of people will always be available for conducting maintenance when the vessel type is used. Ab utilization ratio is used to give the degree to which a vessel type can conduct maintenance tasks in parallel. If this ratio is set to 1, maintenance tasks can be performed in parallel without any time being lost between tasks. If the ratio is set to $1/M_v^{CREW}$, tasks can only be done in sequence. The utilization ratio can also be seen as a measure of how many hours the crew on a vessel work efficiently, out of the hours the vessel spend at a wind farm.

5.1.5 Vessel Cost Structure

In the mathematical formulation, the cost of long-term chartering a vessel depends on the type of vessel to be acquired, the time the chartering agreement is made, and the length of the lease. Vessels that are able to operate in harsher weather conditions and conduct more complex maintenance tasks are assumed to be more expensive to charter than simpler vessels. Furthermore, long-term chartering of vessels is assumed to have a declining yearly cost with the length of the charter. As an example, a chartering contract of 10 years has a lower yearly charter cost than a similar contract of 5 years. In the model, the chartering costs of different vessel types may vary with the year (strategic node) the charter agreement is made. This is included to account for the introduction of new technology and vessel concepts, and gives the opportunity to model long-term trends in vessel charter rates. For example, one can assume that certain vessels are generally cheaper to charter 15 years from now due to technological advances and a declining price trend in the market. In addition, as vessels get older, the maintenance cost of a vessel generally increase, which may result in lower charter prices for older vessels.

For short-term chartering of vessels, the chartering cost is given by the market price of the vessel type in a season. The short-term charter prices vary with the season and demand in the market, but do not vary with lease length. Also for short-term chartering rates, the model can handle trends showing a decline or increase in charter rates. Furthermore, the model assumes that the decision-maker can decide to short-term charter out vessels from the fleet. Only the vessels that are chartered in to the fleet with a long-term charter contract can be chartered out, and these vessels can only be chartered out on a short-term basis. The revenue from short-term chartering out vessels during a specific season are given by the prices in the market. However, due to commission fees, the revenue retrieved per vessel chartered out is assumed lower than the cost of chartering in the same vessel.

The variable cost of using each vessel type is calculated per man-hour that the vessel type is performing maintenance tasks, and travelling to and from wind farms. The hourly variable cost includes costs related to fuel, crew and labour. The variable costs apply to all vessels in the fleet that are deployed in a given period, both the long-term chartered and the short-term chartered vessels. Fixed costs are, on the contrary, assumed to only apply for the vessels that are chartered with a long-term lease. For the vessels with a short-term lease, the fixed costs are assumed to be included in the charter price of the vessel. The fixed costs includes costs related to insurance and vessel maintenance. The model assumes that the decision-maker does not have any budget restrictions when making the charter decisions.

5.1.6 Wind Farms and Maintenance Tasks

To model that a wind farm can be developed through stepwise increments, each step of the development can be modelled as the addition of a new wind farm located at the same place as a current wind farm. In this way, a large wind farm can be seen as a cluster of smaller wind farms with the same distance to a given harbour or offshore station. At the beginning of the planning horizon, there are uncertainties related to the realization of the planned developments. Both the timing and the size of a stepwise increment of a wind farm may differ from the plan. Due to this uncertainty, the set F_n is used to indicate whether a given wind farm exists in a strategic node n. All turbines in a wind farm are assumed to be of the same type, and hence have the same capacity and generated power output during a day.

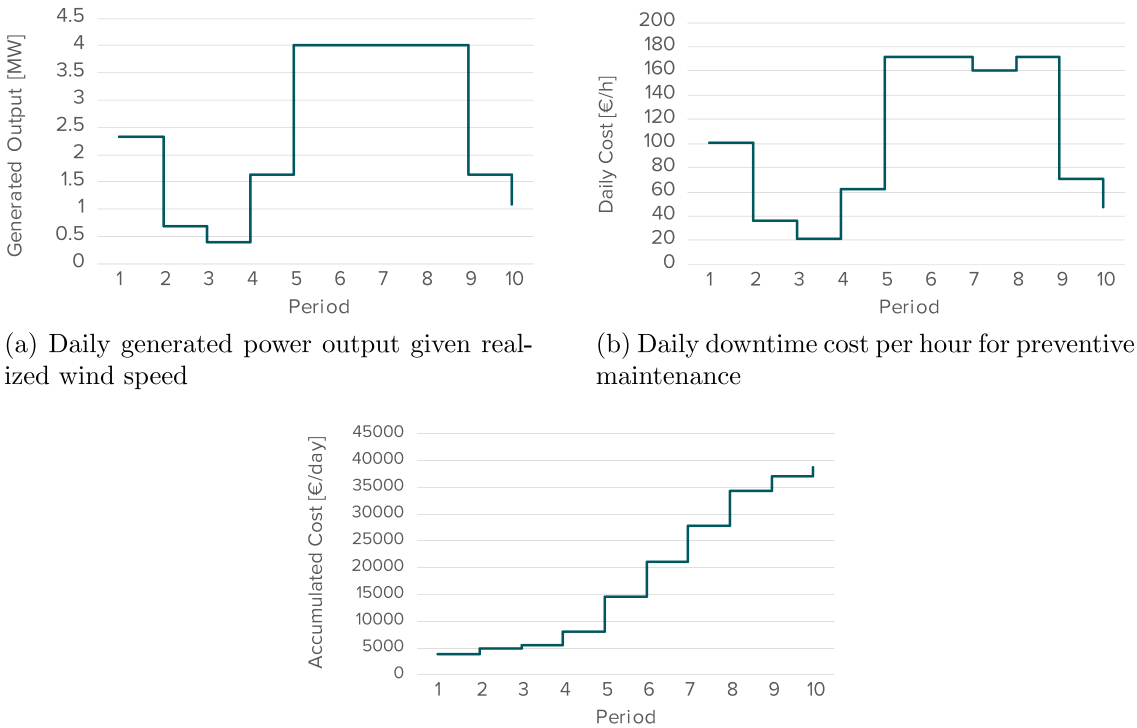
A maintenance task is one specific task which is to be conducted at one wind farm during the duration of a tactical scenario. A specific maintenance task takes a certain number of man-hours to conduct, and requires a certain amount of resources like crew, spare parts and lifting capacity. In addition, every maintenance task occurs in a specific period, which is the point in time when a failure on the turbine appeared. is given with a period of occurrence. A task can be either preventive or corrective. The number of preventive tasks to be conducted at each wind farm is assumed to only depend on the number of turbines at the wind farm, and is assumed known at the beginning of the planning horizon. All preventive tasks hence occur in the first period of a tactical scenario. The total demand for preventive maintenance in each strategic node, however, is uncertain due to the concept of stepwise development of wind farms. The demand for corrective maintenance is not known at the beginning of the planning horizon. This demand is revealed in the beginning of a tactical scenario. Preventive maintenance tasks can be conducted in any period during a tactical scenario, and corrective maintenance tasks can be conducted in any period after its occurrence. However, delaying corrective maintenance with many periods leads to large downtime costs, as described in the following subsection, and corrective maintenance should therefore be conducted as quickly as possible.

Since the demand for corrective maintenance is uncertain, some tactical scenarios may have an abnormally high demand for maintenance. If all maintenance tasks must be conducted, the wind farm owner may be forced to acquire an unrealistically large fleet to be able to fulfill the demand in such scenarios. To avoid this effect, a penalty cost has been introduced, related to not conducting a maintenance task. A binary variable is used to make the constraints that force all maintenance tasks to be conducted soft, and is set to one if a specific task is not conducted. The binary variable and penalty cost is added to the objective function, giving the wind farm owner a trade-off between acquiring a large fleet to be able to fulfill even the highest demand scenarios, or acquiring a smaller fleet, and paying a penalty cost when the fleet renders insufficient for serving all maintenance demand.

5.1.7 Downtime Costs

In the mathematical model, the costs of downtime related to conducting both corrective and preventive maintenance tasks are included. As discussed in Subsection 2.6.1, the main factors that influence downtime costs are: electricity price and subsidy scheme, wind speed, turbine size, turbine efficiency, and time of failure relative to the time of maintenance execution.

For preventive maintenance, the downtime is assumed to be equal to the sum of operational hours that one or several vessel(s) use to conduct a preventive task. It is assumed that it is necessary to shut down the turbine during all kinds of preventive maintenance. The hourly downtime cost of preventive maintenance is given by a step function, using a uniform time-discretisation. Each time interval has the length of one day, implying that wind speeds and electricity prices are assumed constant during one day. The daily generated power output is shown in Figure 5.3(a), and is calculated based on the realized wind speed that day. The daily generated power output of a turbine is used to calculate the hourly downtime cost on a given day, as illustrated in Figure 5.3(b).



(a) Daily generated power output given realized wind speed

(b) Daily downtime cost per hour for preventive maintenance

(c) Accumulated downtime cost for corrective maintenance

Figure 5.3: Example of downtime costs for a 4 MW turbine.

For corrective maintenance, the length of the downtime is modelled as the number of days from the failure occurs until it is corrected. For each day corrective maintenance is delayed, the full downtime cost of that day is incurred. The downtime cost is hence accumulated from the point of failure until the corrective maintenance task is completed, as shown in Figure 5.3 (c). To model the downtime cost of corrective maintenance, the parameter C^{DTC}_{nqspfm} is used. This parameter gives the cost of waiting until period p to complete a corrective maintenance task m, which occurred in an earlier period at wind farm f. A binary variable is used to handle whether a corrective maintenance task is completed in period p or not.

5.2 Definitions

Lower-case letters are used to represent decision variables and indices, while capital letters represent constants and sets.

Indices:

n	Strategic node.
q	Season type.
s	Tactical scenario.
p	Period.
f	Wind farm.
m	Maintenance task.
v	Vessel type.
d	Harbour or offshore station.
l	Lease expiration time.
k	Type of restricting weather condition.

Sets:

N	Set of strategic nodes, $N : \{1, ..., \vert N\vert\}$.
A_n	Set of ancestor nodes of node n, where $A_n \subseteq N$.
Q	Set of season types, $Q : \{\text{Summer}, \text{Winter}\}$.
S_{nq}	Set of tactical scenarios in node n, season q, $S_{nq} : \{1, ..., \vert S_{nq}\vert\}$.
P_{nqs}	Set of periods in tactical scenario s, of node n, season q, $P : \{1, ..., \vert P_{nqs}\vert\}$.
V	Set of vessel types, $V : \{1, ..., \vert V\vert\}$.
V_d	Set of vessel types that belong to a harbour or offshore station d, where $V_d \subseteq V$.
V_m	Set of vessel types able to conduct maintenance task m, where $V_m \subseteq V$.
F_n	Set of wind farms that exist in node n, $F_n : \{1, ..., \vert F_n\vert\}$.
M_{nqsf}	Set of maintenance tasks to be conducted at wind farm f in node n, season q, scenario s, $M_{nqsf} : \{1, ..., \vert M_{nqsf}\vert\}$.
M_{nqf}^{PREV}	Set of preventive maintenance tasks to be conducted at wind farm f in node n, season q, where $M_{nqf}^{PREV} \subseteq M_{nqsf}$.
M_{nqsf}^{CORR}	Set of corrective maintenance tasks to be conducted at wind farm f in node n, season q, scenario s, where $M_{nqsf}^{CORR} \subseteq M_{nqsf}$.
D	Set of harbours and offshore stations, $D : \{1, ..., \vert D\vert\}$.
L_n	Set of possible lease expiration times when chartering is conducted in node n, $L_n : \{t(n+1), ..., t(N)\}$.

L_{nv} Set of possible lease expiration times for vessel of type v being chartered in node n. The lease expiration times are given as the year during the planning horizon in which the vessel lease expires. The vessel will leave the fleet at the beginning of this year. $L_{nv} \subseteq L_n$.

K Set of restricting weather conditions, $K : \{1, ..., |K|\}$.

Parameters:

C^{TC}_{nvl} Total charter cost of a vessel of type v long-term chartered in node n with lease expiration l.

C^{F}_{nv} Fixed cost for vessel of type v in node n.

C^{D}_{d} Total cost of using harbour or offshore station d.

C^{V}_{v} Variable cost per man-hour of operating a vessel of type v.

C^{DTP}_{nqspf} Downtime cost per hour of conducting preventive maintenance at wind farm f in period p, in node n, season q, scenario s.

C^{DTC}_{nqspfm} Downtime cost of corrective maintenance task m at wind farm f, when the task is finished in period p, in node n, season q, scenario s.

C^{P}_{m} Penalty cost of not conducting maintenance task m.

C^{ST}_{nqv} Cost of short-term chartering in a vessel of type v in node n, season q.

R_{nqv} Revenue of short-term chartering out a vessel of type v in node n, season q.

Z_n Discount rate in node n.

B^{S}_{n} Probability of strategic node n.

B^{T}_{nqs} Probability of tactical scenario s in node n and season q.

M^{D}_{d} Maximum capacity at harbour or offshore station d.

G_{vd} Amount of capacity one vessel of type v needs at harbour or offshore station d.

T^{M}_{m} Man-hours needed to perform maintenance task m.

T^{T}_{fv} Transit time of a vessel of type v for travelling back and forth to wind farm f.

T^{MAX}_{v} Maximum operation time for a vessel of type v in one period.

E_v Utilization ratio of man-hours during maintenance execution with a vessel of type v.

M^{CREW}_{v} Maximum size of crew at a vessel of type v.

M^{K}_{vk} Weather capabilities k of a vessel of type v.

U_{nqspk} Value of weather type k in node n, season q, scenario s, period p.

$t(n)$ Time of node n.

$a(n)$ Ancestor of node n.

Decision variables:

x_{nvl} Number of vessels of type v long-term chartered in node n with expiration time l.

w_{nv} Number of long-term chartered vessels of type v in the fleet in node n.

y^{IN}_{nqv} Number of vessels of type v short-term chartered in, in node n, season q.

y^{OUT}_{nqv} Number of vessels of type v short-term chartered out, in node n, season q.

u_{nqspfv} Number of vessels of type v that operates on wind farm f in node n, season q, scenario s, period p.

$t_{nqspfmv}$ Amount of man-hours vessels of type v use to conduct maintenance task m at wind farm f in node n, season q, scenario s, period p.

γ_{nqspfm} Binary variable. 1 if corrective maintenance task m on wind farm f is conducted in node n, season q, scenario s, period p. 0 otherwise.

β_{nqsfm} Binary variable. 1 if maintenance task m occurring at wind farm f, in node n, season q, scenario s is not conducted. 0 otherwise.

δ_d Binary variable. 1 if harbour or offshore station d is used. 0 otherwise.

5.3 The Dual-Level Stochastic Model

A stochastic node formulation of the DLPOW is presented in detail in this section. The objective function is presented first, followed by the constraints. A plain version of the model is presented Appendix A.

5.3.1 Objective Function

$$\min \quad z = \sum_{n \in N} \sum_{v \in V} \frac{B_n^S}{Z_n} \left(\sum_{l \in L_{nv}} C_{nvl}^{TC} x_{nvl} + C_{nv}^F w_{nv} + \sum_{q \in Q} (C_{nqv}^{ST} y_{nqv}^{IN} - R_{nqv} y_{nqv}^{OUT}) \right) \tag{5.1}$$

$$+ \sum_{d \in D} C_d^D \delta_d \tag{5.2}$$

$$+ \sum_{n \in N} \sum_{q \in Q} \sum_{s \in S_{nq}} \frac{B_{nqs}^T}{Z_n} \left(\sum_{p \in P_{nqs}} \sum_{f \in F_n} \left[\sum_{m \in M_{nqsf}} \sum_{v \in V_m} C_v^V t_{nqspfmv} + \sum_{v \in V} C_v^V u_{nqspfv} T_{fv}^T \right. \right. \tag{5.3}$$

$$\left. + \sum_{m \in M_{nqf}^{PREV}} \sum_{v \in V_m} C_{nqspf}^{DTP} t_{nqspfmv} + \sum_{m \in M_{nqsf}^{CORR}} C_{nqspfm}^{DTC} \gamma_{nqspfm} \right] \tag{5.4}$$

$$\left. + \sum_{f \in F_n} \sum_{m \in M_{nqsf}} C_m^P \beta_{nqsfm} \right) \tag{5.5}$$

The objective function z minimizes the total cost of conducting maintenance tasks throughout the planning horizon. The first term of (5.1) gives the total charter cost of all long-term chartered vessels. The second term of (5.1) gives the fixed cost of all vessels that are in the fleet in a given node n. The third term of (5.1) gives the short-term charter costs of the vessels chartered in, while the fourth term gives the short-term charter revenue of the vessels chartered out, in each season. Part (5.1) is discounted with a discount factor to express the present value of future costs, and multiplied with the probability of going through each strategic node. Part (5.2) gives the total cost of using harbours and/or offshore stations. Part (5.3) gives the variable operating cost for all the vessels deployed. The first term gives the variable cost of using vessels to conduct maintenance tasks, while the second term gives the cost of transit between wind farms and vessel origins. Part (5.4) gives the downtime costs due to production stops. The first term gives the preventive downtime cost, while the second term gives the corrective downtime cost. Part (5.5) gives the penalty cost of not conducting maintenance tasks. All the parts (5.3) – (5.5) are discounted and multiplied with the probability of ending up in tactical scenario s of season q and node n.

5.3.2 Constraints for Strategic Nodes

Fleet balance
The number of long-term chartered vessels in the fleet in node n, is the number of vessels long-term chartered in the node, plus the existing fleet from the direct ancestor node $a(n)$, minus the vessels whose lease expire in n. This is handled in Constraints (5.6).

$$\sum_{l \in L_{nv}} x_{nvl} + w_{a(n)v} - \sum_{n' \in A_n} x_{n'vt(n)} = w_{nv}, \qquad n \in N \backslash \{1\}, v \in V \tag{5.6}$$

Constraints (5.7) handle the fact that the decision-maker has no initial fleet in the first node. Hence, in this node, the number of long-term chartered vessels in the fleet is equal to the number of vessels long-term chartered in this node.

$$\sum_{l \in L_{nv}} x_{nvl} = w_{nv}, \qquad n = 1, v \in V \tag{5.7}$$

Chartering out
To ensure that the decision-maker does not charter out vessels she does not possess, Constraints (5.8) state that the number of vessels of type v chartered out in node n, season q, cannot exceed the amount of such long-term chartered vessels available in the fleet in node n.

$$y^{OUT}_{nqv} \leq w_{nv}, \qquad n \in N, q \in Q, v \in V \tag{5.8}$$

Offshore stations
Constraints (5.9) ensure that the capacity required by the vessels in the fleet belonging to harbour or offshore station d does not exceed the total capacity of this harbour or offshore station. If any vessel type belonging to harbour or offshore station d is chartered during the planning horizon, the corresponding harbour or offshore station is used and the cost of using it needs to be accounted for in the objective.

$$\sum_{v \in V_d} G_{vd}(w_{nv} + y^{IN}_{nqv}) \leq M^D_d \delta_d, \qquad n \in N, q \in Q, d \in D \tag{5.9}$$

5.3.3 Constraints for Tactical Scenarios

Maintenance demand
During the duration of a strategic node, the demand for preventive and corrective maintenance tasks must be met. Constraints (5.10) ensure that if a maintenance task is conducted, the number of man-hours used to conduct the maintenance task must be at least the number of man-hours required to conduct task m. If not, the task is assumed to not be conducted at all.

$$\sum_{p \in P_{nqs}} \sum_{v \in V_m} t_{nqspfmv} \geq T^M_m (1 - \beta_{nqsfm}), \qquad n \in N, q \in Q, s \in S_{nq}, f \in F_n, m \in M_{nqsf} \tag{5.10}$$

Operation time
The amount of man-hours that a vessel of type v operates on a wind farm f during a period, must be smaller than the maximum man-hours available for this vessel type during one period. The maximum man-hours for a vessel type is calculated as the utilization ratio of man-hours multiplied with the crew capacity of the vessel type, multiplied with the maximum operation time for the vessel, minus the transit time of the vessel to wind farm f. As several vessels of the same type may operate on a wind farm, the maximum operation time of a vessel type is multiplied by the number of vessels of that type operating on a wind farm in a period. Constraints (5.11) handle this.

$$\sum_{m \in M_{nqsf}} t_{nqspfmv} \leq E_v M_v^{CREW} (T_v^{MAX} - T_{fv}^T) u_{nqspfv},$$
$$n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, v \in V_m. \quad (5.11)$$

The sum of all vessels operating at a wind farm f in period p, in addition to all vessels not in use, needs to be equal to the size of the fleet. Constraints (5.12) determine the number of each vessel of type v located at each wind farm, or at its origin 0, in each period.

$$\sum_{f \in \{0\} \cup F_n} u_{nqspfv} = (w_{nv} + y_{nqv}^{IN} - y_{nqv}^{OUT}), \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V \quad (5.12)$$

Weather
Constraints (5.13) ensure that no vessel type can operate in a period if its weather capabilities are exceeded by the realised weather conditions in that period.

$$(M_{vk}^K - U_{nqspk}) \sum_{m \in M_{nqsf}} \sum_{f \in F_n} t_{nqspfmv} \geq 0, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V_m \quad (5.13)$$

These constraints can be handled by prepossessing the input data, but are given here explicitly for completeness of the model.

Timing of corrective maintenance
For corrective maintenance tasks, the downtime costs are calculated based on when the tasks occur and when they are completed. Constraints (5.14) find the time at which each maintenance task is completed. The constraints ensure that when γ_{nqspfm} related to task m is set to 1 in a period, the task has been completed and no further time can be spent on the task in the following periods.

$$\sum_{p' = \{(p+1), \ldots, |P_{nqs}|\}} \sum_{v \in V_m} t_{nqsp'fmv} \leq T_m^M (1 - \gamma_{nqspfm}),$$
$$n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, m \in M_{nqsf}^{CORR} \quad (5.14)$$

Constraints (5.15) give the convexity constraints for γ_{nqspfm}, and ensure that if a corrective maintenance task is conducted, it is completed in a specific period. If this is not the case, the task is not finished, and the variable β_{nqsfm}, which enforces the penalty cost, is set to 1.

$$\sum_{p \in P_{nqs}} (\gamma_{nqspfm} + \beta_{nqsfm}) = 1, \qquad n{\in}N, q{\in}Q, s{\in}S_{nq}, f{\in}F_n, m{\in}M_{nqsf}^{CORR} \tag{5.15}$$

Non-negativity, integrality and binary constraints

Constraints (5.16) – (5.24) impose non-negativity, integrality, and binary constraints on the respective decision variables. As all chartering is done in strategic nodes, Constraints (5.16) – (5.19) and (5.24) apply to strategic nodes. The remaining consraints apply to the tactical scenarios. For short-term charter, only vessels available in the market can be charted. This is handled by constraints (5.18).

$$x_{nvl} \geq 0 \quad \text{and integer}, \qquad n{\in}N, v{\in}V, l{\in}L_{nv} \tag{5.16}$$

$$w_{nv} \geq 0 \quad \text{and integer}, \qquad n{\in}N, v{\in}V \tag{5.17}$$

$$y_{nqv}^{IN} \geq 0 \quad \text{and integer}, \qquad n{\in}N, q{\in}Q, v{\in}V | L_{nv} \neq \emptyset \tag{5.18}$$

$$y_{nqv}^{OUT} \geq 0 \quad \text{and integer}, \qquad n{\in}N, q{\in}Q, v{\in}V \tag{5.19}$$

$$u_{nqspfv} \geq 0 \quad \text{and integer}, \qquad n{\in}N, q{\in}Q, s{\in}S_{nq}, p{\in}P_{nqs}, v{\in}V, f{\in}F_n \cup \{0\} \tag{5.20}$$

$$t_{nqspfmv} \geq 0, \qquad n{\in}N, q{\in}Q, s{\in}S_{nq}, p{\in}P_{nqs}, v{\in}V_m, f{\in}F_n, m{\in}M_{nqsf} \tag{5.21}$$

$$\gamma_{nqspfm}{\in}\{0,1\}, \qquad n{\in}N, q{\in}Q, s{\in}S_{nq}, p{\in}P_{nqs}, f{\in}F_n, m{\in}M_{nqsf}^{CORR} \tag{5.22}$$

$$\beta_{nqsfm}{\in}\{0,1\}, \qquad n{\in}N, q{\in}Q, s{\in}S_{nq}, p{\in}P_{nqs}, f{\in}F_n, m{\in}M_{nqsf} \tag{5.23}$$

$$\delta_d{\in}\{0,1\}, \qquad d{\in}D \tag{5.24}$$

Chapter 6

Solution Methods

Several examples from the literature show that exact solution methods often are incapable of solving large problems instances of complex stochastic optimization problems, and that approaches based on metaheuristics often outperform exact solution methods for these types of problems. The DLPOW is formulated with a dual-level stochastic modelling approach, which results in a rapid growth in problem size. For this reason, it is expected that the use of a standard optimization solver directly on the mathematical formulation only is applicable for small instances. In order to solve real-life instances of the problem within a reasonable amount time, a metaheuristic has been developed for the DLPOW. In this chapter, the metaheuristic and its main features are presented.

6.1 A Metaheuristic for the DLPOW

The metaheuristic developed to solve the DLPOW is a Greedy Randomized Adaptive Search Procedure (GRASP), with an embedded Greedy Tactical Heuristic, that takes advantage of the structural properties of the problem. The heuristic developed for DLPOW is mainly inspired by the previous work of Pantuso et al. [106], Resende et al. [117], Hvattum et al. [74] and Prais et al. [111].

GRASP is an iterative metaheuristic for solving combinatorial optimization problems. Traditionally, each iteration consists of two phases: construction and local search [117]. In the construction phase, a feasible solution is built from scratch through making greedy choices. In order to generate different solutions in each iteration, the procedure includes some randomness when making these greedy choices. During the construction of a solution, the possible choices that can be made are updated as the solution evolve. This makes the procedure adaptive. In the local search phase, the neighborhood of the constructed solution is investigated until a local minimum is found [117]. In the GRASP developed for the DLPOW, the local search phase has been left out. As mentioned in Chapter 4, Prais et. al propose an extension of the basic GRASP, called Reactive GRASP, in which one of the basic parameters used in the construction phase is self-adjusted with respect to historically good values. Inspired by this, a reactive extension has been implemented in the GRASP developed for DLPOW. The embedded Greedy Tactical Heuristic is based on the concept of a simple greedy heuristic. A greedy heuristic is a construction heuristic that constructs solutions based on a myopic evaluation criteria [73].

As presented in Chapter 4, Pantuso et al. show that if a dual-level stochastic model has a certain structural property, the resulting stochastic program has block-separable recourse [106]. According to Pantuso et al., the property of block-separable recourse is beneficial as it allows treating the multistage dual-level model as a two-stage stochastic program, which can be decomposed into a master problem (MP) and many independent LP subproblems. This facilitates the isolation and reduction of the complicating mixed-integer component of the problem [106]. As discussed in Section 5.1, one of the basic assumptions for the DLPOW is that no tactical decision influences future decisions. Hence, the structural property presented by Pantuso et al. [106], stating that it must be possible to distinguish a set of decisions that have no influence on any other future decisions, holds for the DLPOW.

Inspired by Pantuso et al. [106], the DLPOW has been decomposed into a MP and many independent subproblems. The block-separable structure of the problem is illustrated in Figure 6.1. The MP of the DLPOW is the problem of deciding the cost optimal fleet size and mix in all strategic nodes, including seasonal decisions. The MP consists of Objective function terms (5.1)-(5.2) and Constraints (5.6)-(5.9) presented in Chapter 5. Constraints (5.6), shown in the figure with a dark blue rectangle, are complicating constraints that bind all nodes together. Constraints (5.7)-(5.9), shown in light blue, are separable for each node. Constraints (5.7) only applies to the root node. The subproblem in the DLPOW is the problem of deploying the available fleet in a given scenario at the minimal cost, and includes Objective function terms (5.3)-(5.5) and Constraints (5.10)-(5.15). All constraints in the subproblem are separable for each scenario, resulting in one subproblem for each scenario as illustrated with grey rectangles in Figure 6.1.

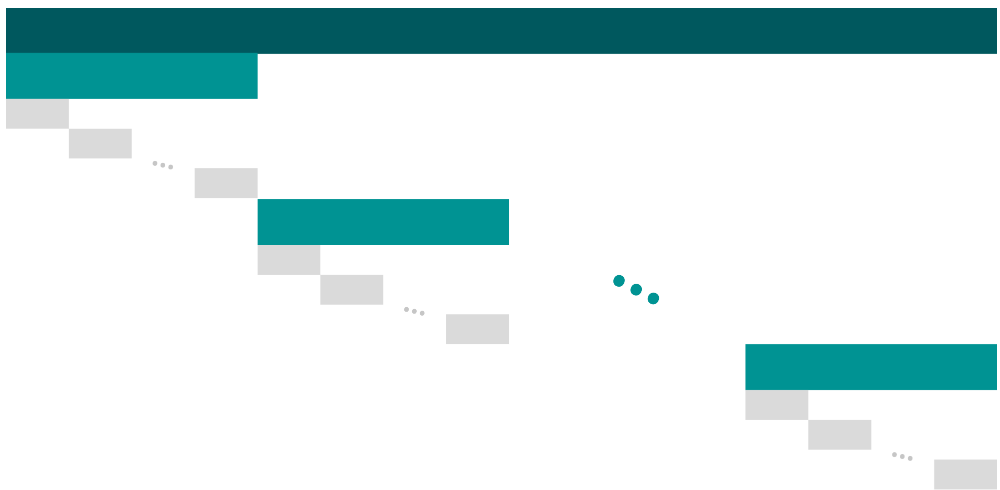


Figure 6.1: Block-Separable Structure of the DLPOW.

As each tactical scenario is independent, the decomposition allows solving each tactical scenario separately for a given fleet size and mix solution of the MP. In order to take advantage of this structural property, the construction phase of the GRASP builds solutions to the MP, while an independent Greedy Tactical Heuristic is embedded in the GRASP to solve the subproblems.

The main procedures of the GRASP for DLPOW, and the flow between the procedures are outlined in Figure 6.2. The `Reactive_GRASP` procedure is executed with a given set of calibration parameters and an initial solution `Initial_Fleet`. In each iteration, the procedure `Greedy_Randomized_Construction` is executed with the given calibration parameters, a randomly drawn α-value, and an initial fleet solution. The `Greedy_Randomized_Construction` starts from the initial solution given, and constructs a fleet size and mix solution iteratively by evaluating all feasible changes to the current partial solution. In each iteration, a specialized set called the reduced candidate list (RCL) is constructed, containing a subset of feasible and improving changes to the current partial solution, and the current partial solution is updated by drawing one element randomly from the RCL. The procedure `Greedy_Tactical_Heuristic` is executed once for each feasible change to the current solution, in order to calculate the objective function value of deploying the fleet that results from making this change in the current partial solution. When the RCL becomes empty, the constructed solution is stored as the best solution if it is better than the best solution found so far. The `Reactive_GRASP` procedure ends when `Greedy_Randomized_Construction` has been executed `Max_Iterations` number of times. The best solution from all iterations is returned upon termination.

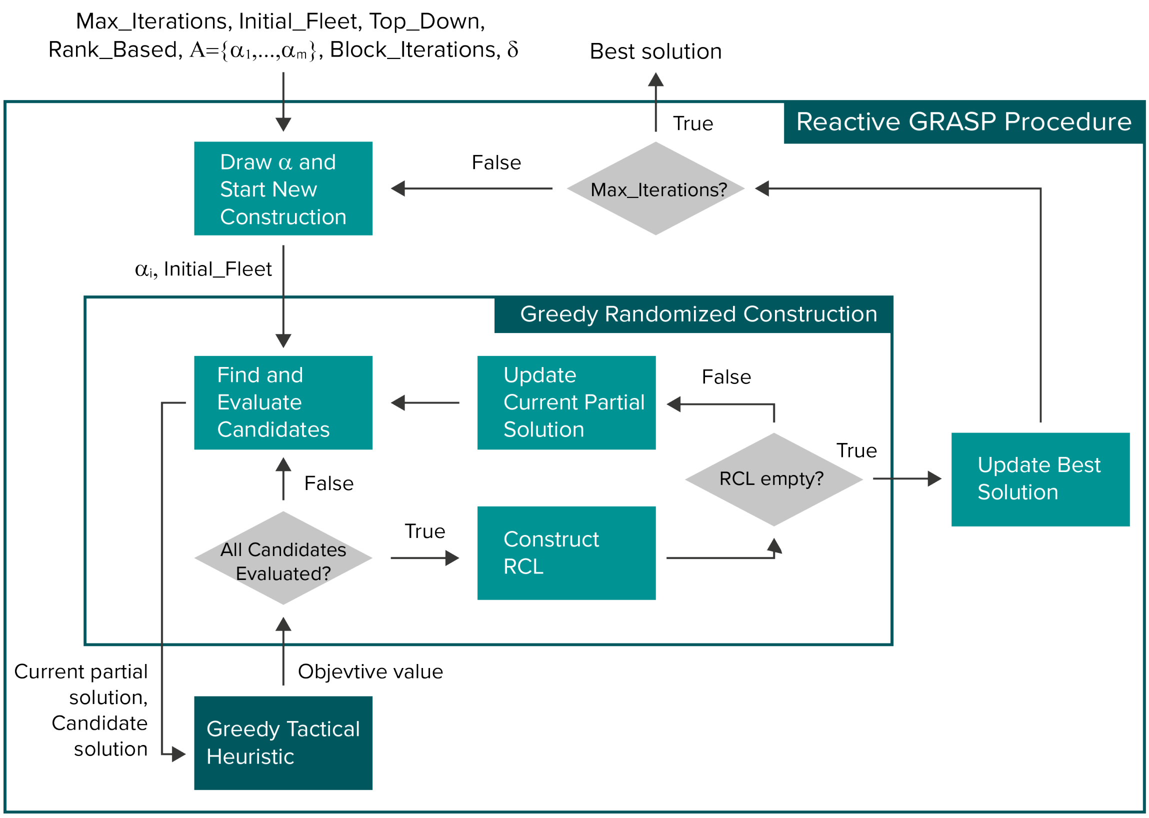


Figure 6.2: Flow of GRASP heuristic developed for DLPOW.

The procedures of the GRASP are described in detail in the remaining parts of this chapter. In Section 6.2, the overall GRASP procedure `Reactive_GRASP` is explained in detail. Section 6.3 gives a detailed explanation of the construction phase of the GRASP, handled by the procedure `Greedy_Randomized_Construction`. In Section 6.4, the reactive extension of the GRASP is explained, while Section 6.5 presents The Greedy Tactical Heuristic implemented for solving the tactical subproblems. Finally, in Section 6.6, strategies used to increase the computational efficiency of the GRASP are presented.

6.2 The Reactive GRASP Procedure

The main outline of the reactive GRASP developed for the DLPOW is shown in Algorithm 1. The GRASP procedure ensures that construction and evaluation of new solutions are conducted iteratively a given number of times.

```
procedure Reactive_GRASP(Max_Iterations, Initial_Fleet, Top_Down, Rank_Based,
  A = {α_1, ..., α_m}, Block_Iterations, δ)
    Best_Solution = Initial_Fleet;
    Initialize_Alpha(A = {α_1, ..., α_m});
    for iteration k = 1, ..., Max_Iterations do
        i = Draw_Alpha(A = {α_1, ..., α_m});
        if Top_Down = true then
            Current_Partial_Solution = Initial_Fleet;
            for n = 1, ..., |Nodes | do
                Current_Partial_Solution =
                  Greedy_Randomized_Construction(Current_Partial_Solution, α_i,
                  Top_Down, Rank_Based, n);
            end
            Solution = Current_Partial_Solution;
        end
        else
            Solution = Greedy_Randomized_Construction(Initial_Fleet, α_i,
              Top_Down, Rank_Based, 1);
        end
        sum_i = sum_i + Solution;
        iterations_i = iterations_i +1;
        Update_Solution(Solution, Best_Solution);
        if ( k modulo Block_Iterations ) = 0 then
            Update_Probabilities(Best_Solution, A = {α_1, ..., α_m}, δ)
        end
    end
    return Best_Solution;
end Reactive_GRASP
```

Algorithm 1: Overall Reactive GRASP procedure developed for DLPOW.

The procedure Reactive_GRASP is executed with a given number of maximum iterations (Max_Iterations), a set of calibration parameters ($A = \{\alpha_1, ...\alpha_m\}$, Rank_Based, Top_Down, Block_Iterations and δ), and a given initial fleet (Initial_Fleet) from which construction starts. The initial fleet is set to the zero fleet by default. The parameter Best_Solution holds the best fleet size and mix solution found, and the corresponding objective function value. Initially, Best_Solution is set to the initial fleet and the corresponding objective value of deploying this fleet. In each iteration, new solutions are constructed with the procedure Greedy_Randomized_Construction, which is outlined in Algorithm 2. The boolean flag Top_Down determines whether this construction is conducted node for node, or if the construction is conducted once for the entire scenario tree as a whole. If the construction is conducted node for node, the parameter Current_Partial_Solution holds the solution under construction.

In each iteration, `Greedy_Randomized_Construction` is called with an α-value which is drawn with a given probability from the set A. The procedure `Initialize_Alpha` sets the initial probability of drawing each α-value to be $1/m$. The probability of drawing each α-value is later updated every `Block_Iteration` iteration (handled by line 16 - 18). The parameters $\texttt{iterations}_\texttt{i}$ and $\texttt{sum}_\texttt{i}$ hold the number of iterations that a given α_i-value, , has been used, and the sum of the objective function values found in these iterations, respectively. Together, line 3, 5, and 16 - 21 constitute the reactive extension of the GRASP, which is explained more thoroughly in Section 6.4.

The procedure `Update_Solution` compares the new solution found in an iteration (`Solution`), with the current best solution (`Best_Solution`), and updates the parameter `Best_Solution` when a better solution is found. After all GRASP iterations have been executed, the globally best solution is returned as the final result.

6.3 The Greedy Randomized Construction

In the construction phase of the reactive GRASP, a feasible solution is built from scratch through greedy choices. The procedure `Greedy_Randomized_Construction` is executed with a current partial solution (`Current_Partial_Solution`), a RCL parameter (α), two boolean flags (`Top_Down` and `Rank_Based`), and a node (n), which guide the construction of a solution. The main outline of the procedure is shown in Algorithm 2.

```
procedure Greedy_Randomized_Construction(Current_Partial_Solution, α,
  Top_Down, Rank_Based, n)
    Candidates = Valid_Candidates(Current_Partial_Solution, Top_Down, n);
    RCL = Find_RCL(Candidates, α, Rank_Based);
    while RCL ≠ ∅ do
        Select a candidate c randomly from RCL;
        Update Current_Partial_Solution according to c;
        Candidates = Valid_Candidates(Current_Partial_Solution, Top_Down, n);
        for candidate j = 1, ..., |Candidates | do
            Calculate objective value of candidate with
              Greedy_Tactical_Heuristic(Current_Partial_Solution, j);
        end
        RCL = Find_RCL(Candidates, α, Rank_Based);
    end
    Solution = Current_Partial_Solution;
    return Solution;
end Greedy_Randomized_Construction
```

Algorithm 2: The construction phase of the GRASP for DLPOW.

The current partial solution (`Current_Partial_Solution`) gives the starting point from which as solution is constructed. The solution is built iteratively by making changes to the current partial solution. The allowed changes, called candidate insertions, are found by the procedure `Valid_Candidates`. All the candidate insertions are evaluated by calculating the objective function value resulting from incorporating the candidate insertion into the current partial solution. Based on the candidates and their evaluation, a RCL is constructed with the procedure `Find_RCL`, and a candidate insertion is randomly selected from the RCL. The current partial solution is then updated by incorporating the selected candidate insertion. From this point, the construction continues by finding the new valid candidate insertions, given the new current partial solution. When no more feasible and improving changes to the partial solution can be found, the RCL becomes empty, and `Greedy_Randomized_Construction` terminates by returning a fleet size and mix solution to the GRASP procedure `Reactive_GRASP`. The three main operations, conducted in each iteration of the `Greedy_Randomized_Construction`, hence are: (1) finding valid candidate insertions, (2) evaluating valid candidate insertions, and (3) constructing the RCL and selecting a valid candidate insertion. These operations are explained in detail in the following subsections.

6.3.1 Finding Valid Candidate Insertions

In each iteration of `Greedy_Randomized_Construction`, a choice must be made regarding which candidate to insert next in the partial solution under construction. In each iteration, the procedure `Find_Valid_Candidates` finds a candidate list (`Candidates`) that contains all valid candidate insertions. The valid candidates are determined by using a rule stating the allowed changes in the partial solution during one iteration of the construction. A typical rule governing the allowed changes is to assign a value to one variable in each iteration, and keep it fixed for the duration of the current construction [117]. However, as the DLPOW has no upper bound on how many vessels that can be chartered in, and hence has an unbounded solution space, this rule is considered insufficient for the DLPOW as it would lead to an unbounded candidate list. Furthermore, as the typical rule only allows alternating one variable at the time, myopic solutions would initially be chosen, as it would initially seem better to charter many vessels of one type. Hence, solutions composed of different vessel types with different charter lengths would not necessarily be explored. To avoid these problems, a specialized decision rule has been developed for the DLPOW. In each iteration of the construction phase, valid candidates are given by the rule:

Rule 6.1 (Valid Candidates) *Increase one of the strategic decision variables in the current partial solution (x_{nvl}, y^{IN}_{nqv} or y^{OUT}_{nqv}) by 1.*

Due to the decomposition of the DLPOW, only the strategic variables are considered to be a part of the solution under construction. The Rule (6.1) makes the candidate list bounded, and reduce the risk of myopic solution being chosen early in the construction. Only feasible solutions are accepted, and hence candidates violating Constraints (5.8), that restricts the farm owner from chartering out vessels she does not posses, are not accepted.

6.3.2 Evaluating Candidate Solutions

In order select a candidate to incorporate into the current partial solution, the quality of each valid candidate needs to be evaluated. A candidate is usually evaluated by calculating the change in the objective function value that results from incorporating the candidate insertion into the current partial solution [117]. This value hence needs to be calculated for each valid candidate.

The strategic cost components of the objective function for the DLPOW can easily be calculated based on the size and mix of the fleet, and is hence directly given by the candidate and current partial solution. However, the tactical decision variables are not explicitly given by the fleet size and mix decision, meaning that for each candidate insertion, all tactical subproblems need to be solved for the given fleet. To handle this, a Greedy Tactical Heuristic (`Greedy_Tactical_Heuristic`) has been embedded in the GRASP for solving the tactical subproblems. The heuristic solves all tactical subproblems in the entire strategic scenario tree for a given fleet size and mix solution, and returns the total objective function value of acquiring and deploying the given fleet. The procedure `Greedy_Tactical_Heuristic` is outlined in more detail in Subsection 6.5.

6.3.3 Constructing the RCL

When all valid candidates in `Candidates` are evaluated, one candidate has to be chosen to be incorporated into the current partial solution. In a GRASP, this is done by randomly selecting a candidate from a RCL [74]. The RCL is constructed by the greedy evaluation function `Find_RCL`, which takes the set of valid candidates, the RCL parameter α and a boolean flag `Rank_Based` as input. The RCL returned by `Find_RCL` is a subset of `Candidates`, and contains the candidates with the best evaluations, namely those candidates which result in the largest decrement in objective function value if selected. Non-improving candidates, with negative decremental objective function values, are not included in the RCL.

In a GRASP, the size of the RCL is restricted by a RCL parameter $\alpha \in [0.00, 1.00]$. The selection of which candidates to include in the RCL can be decided either by the number of candidates (rank based) or by the quality of the candidates (value based) [74]. The boolean flag `Rank_Based` determines whether a rank based or value based selection is used.

With a rank based selection, the RCL contains the x valid candidates giving the highest decrement in objective function value, where:

$$x = \max[1, \lceil \alpha * |\texttt{Valid\_Candidates}| \rceil] \tag{6.1}$$

With a value based selection, the RCL contains the subset of candidates that give a decrement in objective function value close to that of the best candidate, where closeness is defined by α. Explicitly, the selection criterion in the value based selection for the DLPOW can be stated as:

$$eval_c \geq eval^{MAX} + \alpha * (eval^{MIN} - eval^{MAX}) \tag{6.2}$$

where $eval_c$ is the decrement in objective function value caused by a given candidate insertion c, $eval^{MAX}$ is the maximum positive decremental cost of any feasible and improving candidate, and $eval^{MIN}$ is the minimum positive decremental cost of any feasible and improving candidate.

6.4 The Reactive Extension of the GRASP

The RCL parameter, α, defines the restrictiveness of the RCL. The performance of a GRASP is sensitive to this parameter, as it affects the amount of greediness and randomness in the metaheuristic [117]. From Equation (6.1) and (6.2), it can be seen that if $\alpha = 1.00$, all feasible and improving candidates are included in the RCL, making the GRASP purely random. If $\alpha = 0.00$, the RCL only consists of the best candidate solution(s), and the GRASP is hence purely greedy. Resende et al. present GRASP as a repetitive sampling technique, where each iteration produces a sample solution from an unknown distribution, whose mean and variance are functions of α [117]. A high α-value leads to a bad average solution, but a high variance and a high number of diverse solutions. A low α-value leads to a good average solution, but a smaller part of the solution space is examined, and hence the probability of finding the optimal solution decreases. Furthermore, Resende et al. [117] show that the computational time generally decrease with a decreasing α-value, as the search becomes more greedy and the size of the RCL decreases.

In a basic GRASP, the α-value is kept constant at a fixed value in all iterations of the GRASP. However, Prais et al. [111] show that using a single fixed α-value often hinders finding high-quality solutions that could have been found if another value was used. For this reason, using a fixed α requires extensive calibration efforts in order to find an appropriate α-value. A common variation of GRASP that avoids this problem, is the Reactive GRASP first introduced by Prais et al. in [111]. Examples from the literature show that the reactive approach leads to improvements over the basic GRASP, in terms of robustness and solution quality, due to greater diversification and less reliance on parameter tuning [117]. For these reasons, the GRASP developed for the DLPOW has been made reactive. The reactive extension to the GRASP for DLPOW is implemented based on the approach suggested by Prais et al. in [111], and is explained in the following paragraphs.

In a Reactive GRASP, the α-value is self-adjusted according to the quality of the solutions previously found [111]. Instead of using a constant α-value, a discrete set $A = \{\alpha_1, ..., \alpha_m\}$, containing m predetermined values, is used. In each iteration of the construction phase, an α-value is randomly selected from A. When the GRASP procedure starts, the probability distribution $P(X)$ of selecting a particular α_i is set to the uniform distribution, where $P(X = \alpha_i) = \texttt{probability}_\texttt{i} = 1/m$. Throughout the search, the probabilities are periodically updated to favour α-values that have historically lead to good solutions.

Several approaches on how to periodically update the probability distribution $P(X)$ are suggested in the literature. The strategy chosen for the GRASP developed for DLPOW is based on the strategy used in [111], and is shown in Algorithm 3. The parameter $\texttt{average}_\texttt{i}$, is calculated as the average objective function value found in iterations where α_i is used. For each α_i-value, a weight is calculated based on the best solution found so far in the GRASP, over $\texttt{average}_\texttt{i}$. A high value of the parameter $\texttt{weight}_\texttt{i}$ indicates that a given α_i has resulted in good solutions in previous iterations. The probability of selecting an α_i, $\texttt{probability}_\texttt{i}$, is calculated based on the value of $\texttt{weight}_\texttt{i}$, and α_i values that have given historically good solutions are hence given a higher probability. The exponent δ can be used to attenuate the updated values of the probabilities.

```
procedure Update_Probabilities(Best_Solution, A = {α_1, ..., α_m}, δ)
    totalWeight = 0;
    for i = 1, ..., m do
        average_i = sum_i /iterations_i;
        weight_i = (Best_Solution/average_i )^δ;
        totalWeight = totalWeight + weight_i;
    end
    for i = 1, ..., m do
        probability_i = weight_i/totalWeight;
    end
end Update_Probabilities
```

Algorithm 3: Updating probabilities of all α-values.

6.5 The Greedy Tactical Heuristic

The Greedy Tactical Heuristic embedded in the GRASP is outlined in detail in Algorithm 4. The procedure `Greedy_Tactical_Heuristic` solves all tactical scenarios for a given fleet size and mix solution in a greedy manner, and returns the total objective function value. The fleet is found by incorporating the candidate insertion, c, into the current partial solution (`Solution`). The procedure developed ensures that no tactical constraints from the mathematical formulation, presented in Chapter 5, are violated.

```
procedure Greedy_Tactical_Heuristic(Solution, c)
    Calculate strategic costs in all nodes, Cost =
     Calculate_Strategic_Cost(Solution, c);
    for node n = 1, ..., |Nodes | do
        Find fleet size and mix for node n, W_n = Find_Fleet(Solution, n, c);
        for season q = 1, ..., |Seasons | do
            for scenario s = 1, ..., |Scenarios | do
                Perform corrective maintenance with Corrective_Maintenance();
                Perform preventive maintenance with Preventive_Maintenance();
            end
        end
    end
    return Cost;
end Greedy_Tactical_Heuristic
```

Algorithm 4: Greedy Tactical

The strategic cost of a fleet is calculated by the procedure `Calculate_Strategic_Cost` as: the charter costs of all vessels long- or short-term chartered in all nodes, the revenue from short-term chartering out vessels in all nodes, the fixed costs of all vessels long-term chartered in each node, and the cost of using harbour(s) and/or offshore station(s). This cost is calculated directly based on the candidate and the current partial solution. In order to find the total objective value of a candidate, the choice of how and when to perform maintenance tasks need to be made. For each node, season and scenario, these choices are made in a greedy manner and the related tactical costs are calculated. Several greedy choices need to be made, including the choice of: wind farm, vessel, maintenance task and vessel type.

In each strategic node, the available fleet given by the candidate and current partial solution is found with the procedure `Find_Fleet`. For each scenario, the procedure first considers how to utilize the fleet to perform corrective maintenance tasks in a greedy manner. The preventive maintenance tasks are conducted with the spare capacity that remains in the fleet after corrective maintenance has been conducted. This greedy choice ensures that corrective maintenance tasks, which are more expensive to delay, are always prioritized over preventive maintenance tasks. Algorithm 5 and 6 explains the workings of the procedures `Corrective_Maintenance` and `Preventive_Maintenance`, which are greedy algorithms that handle how preventive and corrective maintenance tasks are conducted, respectively.

6.5.1 Corrective Maintenance

Initially, a set `M_Corr_f` containing the corrective maintenance tasks occurring in the first period, is constructed for each wind farm. These sets are updated dynamically when maintenance is performed and when new periods give new demand for corrective maintenance. A cumulative cost, `C_f`, is calculated for each wind farm, as the sum of the accumulated downtime cost incurred if all tasks remaining in `M_Corr_f` are delayed to the next period.

For each period, all vessels in the fleet that are able to operate under the realised weather conditions are saved in the list `Vessels`. The set `Vessels` is dynamically updated to reflect the amount of vessels that have not yet been used. While there still are unused vessels and unfinished corrective maintenance tasks at any wind farm in the given period, a wind farm and a vessel is chosen in a greedy manner. The selected vessel is sent to the selected wind farm, where it conducts maintenance in a greedy manner. When the vessel has been exhausted, or there are no corrective maintenance tasks left at the selected wind farm, the cumulative cost (`C_f`) of the selected wind farm is updated, and a wind farm and vessel is selected again with a new greedy evaluation. When the total vessel fleet has been exhausted for the current period or all tasks are finished, the sets of maintenance tasks (`M_Corr_f`) are updated for all wind farms, and the tasks occurring in the next period are added. The cumulative costs (`C_f`) are also recalculated to reflect the changes in `M_Corr_f`. The procedure `Corrective_Maintenance` stops when all corrective maintenance tasks are conducted or when the fleet is exhausted in all periods.

```
procedure Corrective_Maintenance()
    for farm f = 1, ..., |Farms | do
        Create set of corrective tasks M_Corr_f occurring in first period;
        Calculate cost of delaying each task in M_Corr_f to next period;
        Calculate cumulative cost C_f of delaying all tasks in M_Corr_f;
    end
    for period p = 1, ..., |Periods | do
        Add all vessels in fleet W_n that can operate in period p to list Vessels;
        while Vessels ≠ ∅ & all sets M_Corr_f ≠ ∅ do
            Select farm with highest cumulative cost f = Greedy_Farm(C_f);
            Select cheapest vessels v = Greedy_Vessel(Vessels, f) for farm f;
            Send vessel v to farm f;
            Add cost of sending vessel v to farm f to Cost;
            while vessel v has more capacity & M_Corr_f ≠ ∅ do
                Select most expensive task m = Greedy_Maintenance(M_Corr_f);
                Assign vessel v to perform maintenance task m;
                Add cost of performing task m to Cost;
                if task m is completed then
                    Remove maintenance task m from M_Corr_f;
                end
                else
                    Update hours left of task m;
                end
                Update capacity left on vessel v;
            end
            Remove vessel v from Vessels;
            Update cumulative cost C_f of remaining tasks in M_Corr_f for farm f;
        end
        if period p ≠ last period then
            for farm f = 1, ..., |Farms | do
                Add all tasks occurring in the next period to M_Corr_f;
                Update cumulative cost C_f of dealying all tasks in M_Corr_f;
            end
        end
        else
            for farm f = 1, ..., |Farms | do
                Add penalty cost of all remaining tasks in M_Corr_f to Cost;
            end
        end
    end
end Corrective_Maintenance
```

Algorithm 5: Corrective Maintenance

The greedy choice of a wind farm is conducted by choosing the wind farm that has the highest cumulative cost of delaying maintenance (`C_f`). The consequence of this greedy criterion is that the algorithm chooses the wind farm with the most remaining man-hours left (at the highest cost). The cheapest vessel for a given wind farm is found by evaluating the maximum hours a vessel can operate on a given day at the wind farm, in relation to the maximum variable cost incurred when using the full capacity of the vessel at that farm:

$$\begin{aligned}
&\frac{\text{Maximum Hours Available}}{\text{Maximum Variable Cost}} = \\
&\frac{\text{Crew} * \text{Utilization Ratio} * (\text{Maximum Operation Time - Transit Time})}{\text{Crew} * \text{Utilization Ratio} * \text{Maximum Operation Time} * \text{Variable Cost}} = \\
&\frac{\text{Maximum Operation Time - Transit Time}}{\text{Maximum Operation Time} * \text{Variable Cost}}
\end{aligned} \tag{6.3}$$

The vessel with the highest ratio is chosen, as this gives the most man-hours available per euro spent. The greedy choice of a corrective maintenance task is made by choosing the task with the highest accumulated downtime cost in the current period, relative to the amount of man-hours needed to finish the task. This means that the greedy algorithm finds a balance between prioritizing tasks that occurred early and tasks that can be finished fast (including small tasks and tasks that have already been started).

The tactical costs incurred from sending vessels to wind farms and from conducting maintenance tasks, are added to the the parameter `Cost` as tactical decisions are made. At the end of the last period, the penalty cost of all remaining tasks in the sets `M_Corr_f` for all wind farms is added to the total cost, as these tasks have not been finished.

6.5.2 Preventive Maintenance

As the downtime cost of preventive tasks only occur while the tasks are actually being conducted, it is lucrative to conduct preventive maintenance in periods with low wind speed, where the hourly preventive downtime cost is low. This differs from corrective maintenance, where it is best to conduct tasks as soon as possible to keep the accumulated downtime cost low. For this reason, the greedy algorithm for conducting preventive maintenance tasks is constructed somewhat different compared to the greedy algorithm for corrective maintenance tasks.

At the beginning of the procedure `Preventive_Maintenance`, a set `M_Prev_f` containing the preventive maintenance tasks occurring in the scenario is constructed for each wind farm. During the procedure, this set is updated dynamically to reflect that tasks have been conducted. Preventive tasks are conducted with the remaining fleet capacity after corrective maintenance tasks have been conducted, either until the total fleet is exhausted in all periods, or until all preventive maintenance tasks have been finished at all wind farms. The algorithm attempts to use the cheapest vessel type, for a given wind farm, as much as possible in all periods. The periods in a scenario are sorted in a greedy manner

```
procedure Preventive_Maintenance()
    for farm f = 1, ..., |Farms | do
        Create set of preventive tasks M_Prev_f to be conducted in scenario s;
    end
    Create list Sorted_Periods with periods sorted from cheapest to most expensive;
    Find spare capacity of all vessels of each type and add to matrix Spare_Capacity;
    while Spare_Capacity ≠ ∅ & all sets M_Prev_f ≠ ∅ do
        Select farm with most preventive maintenance left f = Greedy_Farm_Prev();
        Select cheapest vessel type t = Greedy_Vessel_Type(Spare_Capacity, f);
        Define break point Outer_Loop;
        for period p in Sorted_Periods do
            while M_Prev_f ≠ ∅ do
                for vessel v of type t in Spare_Capacity do
                    if vessel v of type t is located at depot then
                        Add cost of sending vessel v of type t to farm f to Cost;
                    end
                    while vessel v of type t has capacity do
                        Select smallest preventive task m =
                         Greedy_Maintenance(M_Prev_f);
                        Assign vessel v of type t to perform maintenance task m;
                        Add cost of performing task m to Cost;
                        if task m is completed then
                            Remove maintenance task m from M_Prev_f;
                        end
                        else
                            Update hours left of task m in M_Prev_f;
                        end
                        Update capacity on vessel v of type t in Spare_Capacity;
                    end
                    Evaluate whether to change farm with Greedy_Farm_Prev();
                    if new farm f is selected then
                        break Outer_Loop;
                    end
                end
            end
        end
    end
    for farm f = 1, ..., |Farms | do
        Add penalty cost of all remaining tasks in M_Prev_f to Cost;
    end
end Preventive_Maintenance
```

Algorithm 6: Preventive Maintenance

based on the hourly downtime cost of performing preventive maintenance at the wind farm with the largest turbines, from lowest to highest. In this way, the algorithm attempts to utilize the cheapest vessel in the cheapest period first, before it considers using the cheapest vessel in the second cheapest period. The second cheapest vessel is only utilized after all capacity of the cheapest vessel type has been exhausted in all periods.

The greedy choice of wind farm is made by selecting the farm that has most preventive maintenance tasks, with the highest downtime cost, left. The cheapest vessel type is found by the greedy criterion given in Equation 6.3. When a vessel type has been chosen, the algorithm assigns a vessels of this type to the selected wind farm. The algorithm always tries to use vessels of this type that have already been sent to the farm first, before sending a vessel from depot. If the vessel has not already been sent to the wind farm to conduct corrective maintenance, the cost of sending the vessel to the wind farm is added to cost parameter `Cost`. The selected vessel is then used to conduct preventive tasks until its capacity is exhausted or no tasks are left at the wind farm. The preventive maintenance task with fewest hours left at the selected wind farm is conducted first, in order to prioritize completing tasks that have already been started. When the capacity of the vessel has been exhausted, the algorithm re-evaluates the choice of wind farm. If a new wind farm is chosen, the algorithm breaks out of the loop `Outer_Loop`, re-evaluates the vessel type, and starts evaluating the cheapest period again. Similarly as for corrective maintenance, the cost of conducting maintenance tasks and the penalty cost of remaining tasks are added to the parameter `Cost` as tactical decisions are made.

6.6 Strategies for Increasing Efficiency

Several measures have been taken to reduce computational time required by the GRASP. The measures can be categorized into two main strategies: reducing the number of valid candidates and using memory structures to avoid recalculations. These measures are described in the following subsections. In addition, the choice of leaving out the local search phase has been made in order to limit computational effort needed. In general, a local search phase increase the probability of finding high quality solution at the cost of an overhead in computational time. As the GRASP for the DLPOW use a rule in the construction phase that leads to the possibility of constructing all feasible solutions, it is considered likely that the GRASP will manage to find high quality solutions without a local search phase. It is therefore expected that a local search phase will not lead to a significant enough improvement in solution quality, to compensate for the added computational effort.

6.6.1 Reducing the Number of Valid Candidates

Evaluating a candidate insertion is time consuming, as this requires all tactical subproblems to be solved for the given fleet size and mix solution. In each iteration of the construction phase, a large number of possible candidate insertions, $(x_{nvl}, y^{IN}_{nqv}, y^{OUT}_{nqv})$, need to be evaluated. As an example, for a small problem instance with 4 nodes, 4 vessel types and 3 possible lease lengths, the number of candidates that need to be evaluated equals:

$$|N| * |V| * |L| + 2 * |N| * |Q| * |V| = 4 * 4 * 3 + 2 * 4 * 2 * 4 = 112 \qquad (6.4)$$

Considering this, one way of reducing computational effort is to reduce the number of candidate insertions that needs to be evaluated in each iteration of the construction.

To prevent certain low-quality insertions from being considered, an additional rule has been added to the original rule, Rule 6.1, used to find valid candidates. This additional rule prohibits decisions where vessels of the same type are short-term chartered in and out in the same season. Such decisions can be considered irrational, as chartering in a vessel with the sole purpose of chartering it out always leads to an additional cost without any gain. The additional rule is given as:

Rule 6.2 (Short-Term Chartering Rule) *If a vessel of type v is chartered out in season s of node n, no vessels of this type can be chartered in, in this season and node combination. If a vessel of type v is chartered in in season s of node n, no vessels of this type can be chartered out in this season and node combination.*

The original rule, Rule 6.1, allows increasing one strategic decision variable with 1 in each iteration. This rule has mainly been chosen to avoid myopic choices early in the construction, as discussed Subection 6.2. However, this rule also has a positive effect on computational effort, as it restricts the number of candidate insertions. As an example, an allowed increase of 2 would double the amount of candidates in each iteration, and hence increase computational time significantly.

The structure of the problem can also be exploited to reduce the number of valid candidates. As described in Section 4.4, Hvattum et al. [74] exploits the scenario tree structure of their problem in order to reduce the number of candidate solutions. In the top-down version of their GRASP, they limit the valid insertions in a top-down fashion. All decisions made in the root node are considered first, before continuing the construction node by node, recursively, in the tree. Reducing the number of allowed insertions in a top-down fashion is expected to have a significant impact on the computational time required by the GRASP, as this reduces the number of valid candidates with a factor of $|N|$. However, while hopefully being more efficient, the top-down version of the GRASP places extra restrictions on the search space available to the metaheuristic, which may lead to myopic solutions. The starting point for the construction of solutions in the GRASP for DLPOW is always the zero-fleet. This means that at the start of the construction, there are no vessels available to conduct maintenance in any node of the scenario tree. When only considering chartering vessels in the root node first, long-term chartering for the full length of the planning horizon may seem unrealistically good, as this has positive effects for all children nodes in the tree. For this reason, the top-down version of the GRASP may give lower quality solutions, where too many vessels are long-term chartered in the root node. Two different version of the GRASP for DLPOW have therefore been implemented: a top-down GRASP (TDG) and an any-node GRASP (ANG). Having an any-node version of the GRASP gives the option of not restricting the solution space in a top-down down manner, at the expense of increased computational time. The boolean flag `Top_Down`, determines which of the GRASP versions that are used.

6.6.2 Using Memory Structures to Avoid Recalculations

The structure of the DLPOW can also be utilized to increase efficiency in other ways. When evaluating candidate insertions, each possible insertion does not necessarily affect the fleet size and mix in all nodes and seasons of the scenario tree. As an example, if the insertion made in the current partial solution is an increase of short-term charter in summer in node n, only the cost of scenarios during summer in node n differ in the potentially new partial solution, when compared to the current partial solution. Similarly, if the insertion made is a long-term charter in node n, only the tactical costs of scenarios in node n and its children are affected.

To exemplify further, Figure 6.3 illustrates one insertion into a current partial solution for a problem instance with two vessel types and seven nodes. To the left in the figure, the current partial solution and the resulting fleet size and mix in each node is shown. The partial solution resulting after one candidate insertion is illustrated to the right in the figure. The insertion exemplified is the increase of the decision variable $x_{2,2,15}$, meaning that the amount of vessels of type 2, with a lease length that expires in year 15, is increased with 1 in node 2. As shown in the figure, only node 2, 4 and 5 are affected by the insertion, and hence only scenarios belonging to these nodes need to be solved to evaluate this candidate insertion.

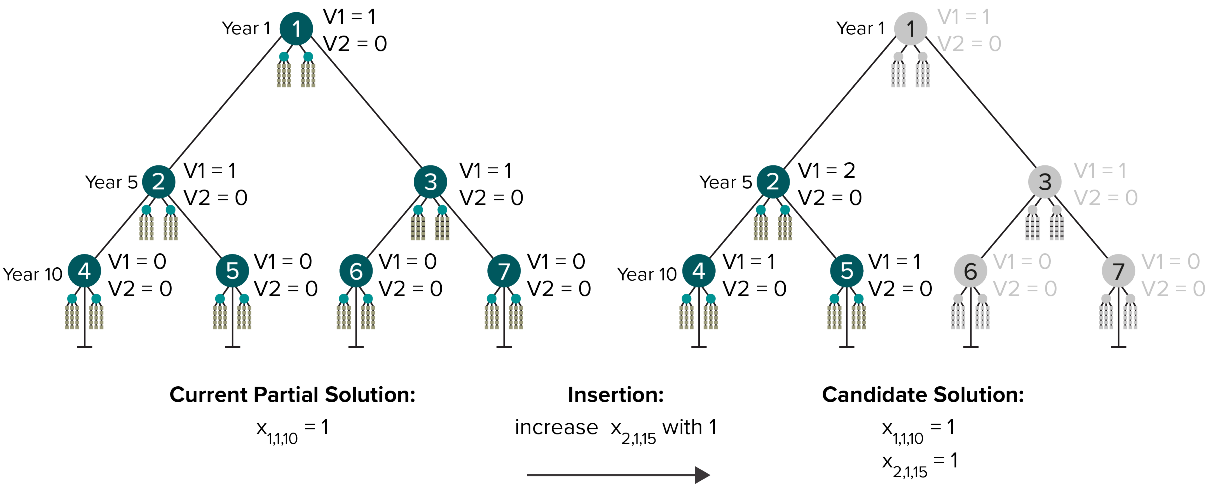


Figure 6.3: Candidate insertion into a current partial solution.

Recalculations can also be avoided by considering similarities in strategic solutions. When deploying the fleet in tactical scenarios, the strategic decision of how to acquire vessels does not affect the tactical costs. Only the total number of vessels of each type available in a tactical scenario affect the tactical costs, as this restricts the deployment of the fleet. Various fleet size and mix decisions, which results in the same fleet size and mix, but differ in how vessels are acquired, are therefore considered equivalent from a tactical point of view. As an example, in a case with two nodes, two strategic solutions that are equivalent from a tactical perspective are: (1) long-term chartering 3 vessels of type 1 for the whole planning horizon in the root node, and (2) short-term chartering 3 vessels of type 1 in both seasons of each node.

To avoid resolving tactical scenarios where the cost of deploying a specific fleet has already been calculated, a hashing-based data structure is utilized. The data structure stores the tactical cost of deploying a given fleet in all scenarios belonging to a given node and season combination. When the Greedy Tactical Heuristic is executed to evaluate a specific candidate, the hashing-based data structure is checked to determine whether the tactical scenarios have been solved previously. In this way, the Greedy Tactical Heuristic avoids recalculation of tactical costs both due to the strategic nodes that are not affected by the insertion under consideration, and due to the different strategic solutions which are equivalent from a tactical perspective.

Chapter 7

Method of Computational Study

The solution methods for the DLPOW have been implemented in two different computer software. The mathematical model presented in Chapter 5 has been implemented in Mosel trough the commercial optimization software FICO(TM) Xpress-IVE. A scenario generator and the metaheuristic presented in Chapter 6, have been implemented in Java through the Java Integrated Development Environment (IDE) from Eclipse.

Static input parameters provided by users require preprocessing and calculation before they are used by the optimization software or by the metaheuristic. Input parameters are given by the user in an Excel file. This data is read by a scenario generator which preprocesses data, calculates parameters and generates tactical and strategic scenarios. The output from the scenario generator is a text file containing an instance of the strategic fleet size and mix problem. The text file is provided as an input file to the optimization software and to the metaheuristic, and includes all sets and parameters required to run the mathematical model. Figure 7.1 shows the interaction between the Excel file, the scenario generator, the problem instance, the optimization solver and the metaheuristic.

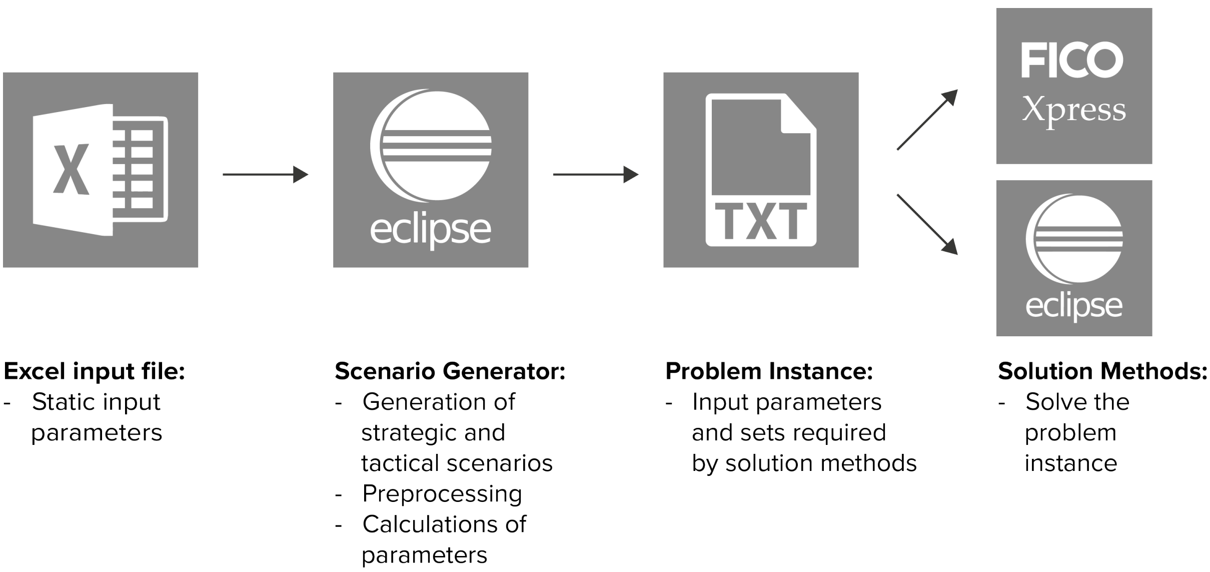


Figure 7.1: Interaction between components used in the computational study.

The choice of static input parameters used in the computational study is described in Section 7.1. The scenario generator developed to create test instances is presented in Section 7.2. The calculation of input parameters and sets required by the mathematical model is described in Section 7.3. Section 7.4 discuss methods to evaluate the implemented scenario generator, while Section 7.5 discuss methods to evaluate the value of including uncertainty in strategic parameters in the mathematical model.

7.1 Selection of Input Data

In order to simulate realistic problem instances for the DLPOW, input data used in the computational study is been based on previous research from the offshore wind industry. This section describes the deterministic data used to generate test instances for the computational study in Chapter 8.

7.1.1 Vessels, Harbours and Offshore Stations

The mathematical model can consider the use of many different vessel types, ranging from helicopters and jack-up barges to CTVs. In addition, the model can separate between vessel types available in the market today, and vessel types that may become available in the future. For simplicity, only CTVs have been used in the computational study. This implies that no heavy-lifting can be done by any of the vessels, and for the computational study it is hence assumed that no maintenance tasks require any heavy-lifting. The input data related to vessel types are based on Gundegjerde et al.[64], Vefsnmo [134], Dinwoodie [43] and Dalgic [39]. The operational characteristics of the vessels types used in the computational study are shown in Table 7.1. Both vessel types are assumed to be available in the market from the beginning of the planning horizon. The capacity values are given relative to a harbour or offshore station's maximum capacity, and do not reflect any real world unit of measure. Table 7.1 does not extensively show all vessel types used in the computational study, as a vessel type is defined uniquely for the harbour or offshore station it belongs to.

Table 7.1: Operational characteristics of different vessels types.

Vessel Number	Vessel Type	Crew [#people]	Speed [knots]	Wave Limit [m]	Wind Limit [m/s]	Utilization [%]	Capacity [-]
1	CTV (small)	12	20	1,5	20	0.7	2
2	CTV (large)	26	18	2,5	25	0.6	3

As mentioned in Section 2.6.2, the cost of acquiring a vessel varies with its type and operational characteristics. Furthermore, the acquisition costs are influenced by other factors, such as the length of the charter agreement, seasonality and the availability of vessels in the market. While an offshore wind farm owner is likely to have good access to vessel cost data, such data is not easily available from any public source. However, the main purpose of the computational study in this thesis is to test the mathematical model and solution methods, rather than to analyze costs. With this in mind, the cost parameters' size relative to each other is more important than the accuracy of the independent cost parameters. For this reason, all vessel cost parameters used in the computational study are estimated based on data from Gundegjerde et al. [64], Dinwoodie [43] and Kaiser [79].

Cost parameters for the vessel types, outlined above, are presented in Table 7.2. The costs of long-term chartering vessels are presented in the column labeled "1 year", and the values give the costs of chartering vessels for one year in the beginning of the planning horizon. The 1 year charter rates are used to calculate the total long-term charter rates relative to the lease lengths. The charter rates may vary between strategic nodes to reflect long-term trends. Short-term charter rates and revenues are given per season, and are assumed to be higher in summer than in winter. The costs are estimated in such a manner that it is never profitable to charter in vessels for the sole purpose of chartering them out.

Table 7.2: Main cost parameters of different vessel types.

Vessel Number	1 year [m€/yr]	Summer(in) [m€]	Summer(out) [m€]	Winter(in) [m€]	Winter(out) [m€]	Fixed [m€/yr]	Variable [€/h]
1	1.1	0.72	0.33	0.69	0.23	0.10	300
2	1.8	1.17	0.54	1.12	0.44	0.15	600

The harbours and offshore stations used in the computational study are illustrated in Table 7.3. The total cost of a station includes all fixed and variable costs related to acquisition and use, and is given for the whole planning horizon. The distances to the various wind farms are given in nautical miles. All harbours are assumed to be located onshore, and are assumed to have larger capacity and lower costs than offshore stations.

Table 7.3: Characteristics of harbours and offshore stations.

Station Number	Station Type [-]	Total Cost [m€]	Max Capacity [-]	Distance to Farm 1 [nmi]	Distance to Farm 2 [nmi]	Distance to Farm 3 [nmi]
1	Harbour	36	100	90	90	50
2	Offshore Station	40	80	70	70	50
3	Offshore Station	38	60	60	10	80

7.1.2 Wind Farms and Maintenance Tasks

The input parameters for wind farms are based on widely used turbine technology developed by Siemens [126], and is presented in Table 7.4. The column "Year" indicates the number of years until the wind farm is planned to be fully realized, relative to the start of the planning horizon. Table 7.4 illustrates a case where two co-located wind farms, with a total of 200 turbines, exist from the beginning of the planning horizon, while one larger wind farm is planned to be realized after 10 years. The wind farm planned to be realized in year 10 has been given larger wind turbines to reflect turbine technology trends, as discussed in Section 2.2. All wind farms are assumed to have turbines with the same cut-in and cut-out speed, at 5 m/s and 25 m/s, respectively.

Table 7.4: Characteristics of wind farms.

Wind farm	Year	Number of Turbines	Capacity [MW]	Swept Area [m^2]
1	0	100	4	13,300
2	0	100	5	13,300
3	10	100	6	18,600

Input parameters related to maintenance tasks are illustrated in Table 7.5, and is based on data from [43]. Each failure rate is given as the average number of failures per turbine per year. Preventive maintenance is scheduled to be conducted once a year per turbine, and it is assumed that no preventive maintenance is conducted during winter. Random failures have been divided into four different types, with different failure rates and resource requirements, leading to four types of corrective maintenance tasks. In order to simplify, resource requirements related to heavy-lifting and supplement of spare parts have not been included.

Table 7.5: Characteristics of maintenance tasks.

Maintenance Type	Type of Task	Repair Time [#man-hours]	Minimum Technicians [#people]	Failure Rate [#failures/yr]
Preventive	Annual service	180	3	1
Corrective	Major repair	104	4	0.04
Corrective	Medium repair	66	3	0.275
Corrective	Minor repair	15	2	3.0
Corrective	Manual reset	6	2	7.5

7.2 Scenario Generation

In this section, the scenario generator developed for the DLPOW is explained in detail. When considering the uncertain parameters in the problem, it is difficult to determine whether one realization of uncertainty is more likely to occur than another. For this reason, all strategic scenarios are considered to have equal probabilities. The same applies for tactical scenarios.

7.2.1 Generation of Tactical Scenarios

The tactical uncertainty is related to two parameters: weather conditions and demand for corrective maintenance. For each tactical scenario, the scenario generator randomly generates values for these parameters, by sampling from historical weather data and a Poisson distribution of the number of failures. Correlation between weather conditions and demand for corrective maintenance has not been considered.

The mathematical model can account for several types of weather conditions, such as wind speed, wave height, ocean current and wind direction. However, for simplicity, only wave height and wind speed are considered in the scenario generator. To generate weather scenarios, historical data from an offshore platform in the Central North Sea [43] has been used. To ensure that seasonal trends in weather conditions are accounted for, the historical data is separated into two sets, representing data for summer and winter separately. The weather scenarios are generated by sampling historical data for all tactical scenarios, and assigning sampled wind speed and wave height to each period in each scenario. To ensure correlation between weather conditions at consecutive days, the sampling method draws a weather realization for the first period randomly from the historical data, while the rest of the periods in the scenario are assigned weather realizations from the consecutive days in the dataset.

The generation of corrective maintenance demand is based on yearly failure rates given by Dinwoodie [43], presented in Table 7.5. The average failure rate, of each failure type e at one farm in one scenario, is calculated as:

$$\lambda_e = \frac{A_e * |Turbines_f| * |Periods|}{365} \tag{7.1}$$

where A_e is the yearly average failure rate of failure type e, $|Turbines_f|$ is the total number of turbines at farm f, and $|Periods|$ is the number of periods in the scenario.

In every scenario, a random number of failures for each failure type, R_e, is drawn for each farm based on the value of λ_e. The random number R_e is drawn from a Poisson distribution expressing the probability of a given number of failures of a type occurring at the farm within the length of the scenario:

$$f(k; \lambda_e) = \Pr(R_e = k) = \frac{\lambda_e^k * e^{-\lambda_e}}{k!} \tag{7.2}$$

When the number of failures for each type, R_e, has been randomly generated, the occurrence of each failure in the scenario is generated randomly. The period in which a failure occurs is drawn from a uniform distribution for each failure at each farm.

As discussed in Subsection 2.3.2, the average failure rates of a turbine often follow a bathtub curve, with a higher number of failures in the early and late stages of a turbine's lifetime. The scenario generator can easily incorporate the bathtub curve, by calculating the average failure rate for each strategic node separately. The average failure rate is then given for each node and farm, and adjusted to reflect the age of the wind turbines. However, the bathtub curve has not been included in the failure rates for the computational study of this thesis.

7.2.2 Generation of Strategic Scenarios

The data used to generate strategic scenarios are given by the user, and no randomness is introduced in the strategic scenario generation. The scenario generator considers uncertainty related to long-term trends in electricity price and stepwise development of wind farms. Other uncertainties, such as uncertainties in governmental subsidy schemes, vessel technology, and fluctuations in vessel charter rates, can also be handled by the the mathematical model, but have not been implemented in the scenario generator. As the problem grows rapidly in size with an increasing number of strategic nodes, strategic scenarios are generated for one type of strategic uncertainty at the time. This allows relatively small instances to be generated, that are expected to be solved within reasonable time.

Uncertainty in Electricity Price

Uncertainty in long-term trends of electricity prices is modelled as a Markov chain, where the electricity price in a strategic node is given with a deviation from the electricity price in its direct parent node. The user gives an initial electricity price for each season in the first node of the planning horizon, which is used to set the electricity price in all following nodes. Each strategic scenario represents a percentage deviation from the electricity price in the direct ancestor node. The electricity price in a node hence depends on the initial electricity price given for the root node and the electricity prices in all previous ancestor nodes. The percentage deviations in each strategic scenario is given by the user. An example of a scenario tree with uncertainty in electricity price is illustrated in Figure 7.2.

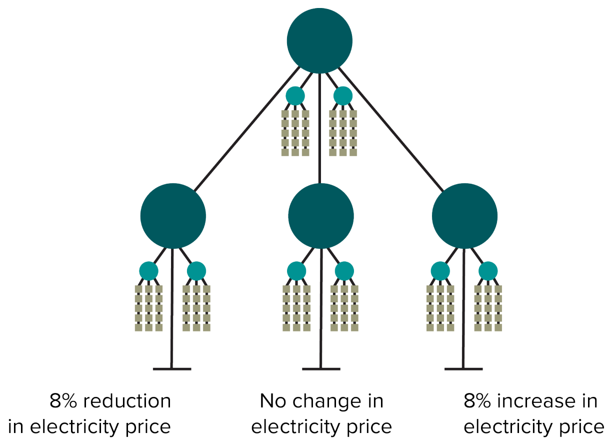


Figure 7.2: A scenario tree with uncertainty in long-term trends of electricity prices.

Uncertainty in Stepwise Development of Wind Farms

The strategic uncertainty related to stepwise development of wind farms considers whether a wind farm development project is fully realized in the planned year or not. The strategic scenarios indicate how many of the planned number of turbines that are installed and operative in a given year. As an example, if a wind farm project with a total of 100 turbines is planned to be fully realized in year 10, three scenarios for year 10 could be that: 0 turbines are installed (0% of the planned number), 50 turbines are installed (50% of the planned number) or all 100 turbines are installed (100% of the planned number). In the case where 0 or 50 turbines are installed, the remaining turbines could still be installed in a later period. Figure 7.3 shows the scenario tree for this example. More stages could be added in the scenario tree to allow delaying the project further.

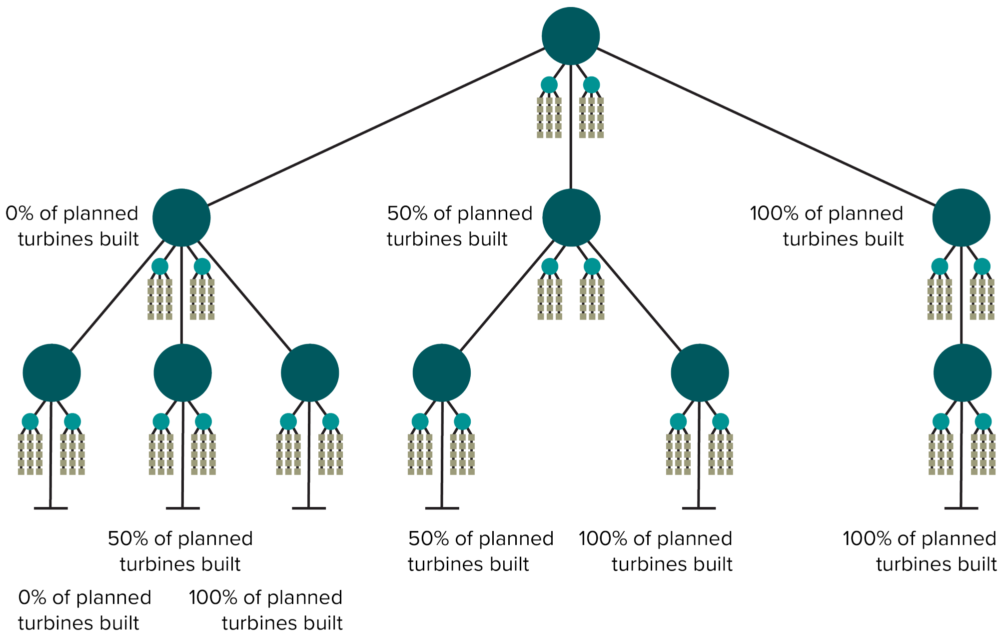


Figure 7.3: A scenario tree with uncertainty in stepwise development of wind farms.

The scenario generator creates the strategic scenarios based on input from the user. The user gives the branching factor of the root node, and the percentages of the planned number of turbines that may be installed in a certain year. In addition, the user states the number of strategic stages in the tree, consequently giving the number of times that the wind farm development can be delayed. To obtain the tree given in Figure 7.3, the user would give the input: 3 stages, branching factor = 3, and the percentages $\{0\%, 50\%, 100\%\}$. In general, if a strategic node n in a tree where the root node has M children and $x_i \in \{x_1, ..., x_i, ..., x_M\}$ of the planned turbines are operative in n, then node n has $|\{x_i, ..., x_M\}|$ children.

7.3 Calculation of Parameters

In order to create test instances that can be used by the solution methods for DLPOW, calculations need to be conduced on the static input data provided by users. This section outlines the calculation of important input parameters, and some simple preprocessing handled by the scenario generator.

7.3.1 Structure of the Scenario Tree

The structure of the scenario tree is generated based on calculations on user input. The user provides input regarding: the number of strategic stages in the scenario tree, the number of strategic scenarios, the duration of time between the strategic stages, the number of tactical scenarios, and the number of periods in each scenario. All parameters and sets related to the structure of the scenario tree, such as the set of ancestor nodes and the set of possible lease lengths, are created based on this input. The mathematical model can include all four seasons of the year, but for simplicity only summer and winter are considered in the computational study, where each season is assigned a length of 6 months. The number of tactical scenarios per season and the number of periods in each scenario are assumed to be the same throughout the scenario tree, even though the mathematical formulation allows variations in these parameters. These simplifications are done to avoid excessive user input and reduce the risk of mistakes in user input as the tree grows large.

The tactical level in the DLPOW has a daily time resolution. If each scenario was to be represented with the number of periods equal to the actual number of days in the duration of the strategic node, the problem would grow rapidly in size. As an example, if the duration of a strategic node is 5 years, a scenario occurring during either summer or winter would have $\frac{365}{2} * 5 = 912.5$ periods. As this would be computationally expensive to represent, the scenarios are represented by including only a fraction of the periods. The costs occurring in each scenario are adjusted with a multiplier to estimate the full cost throughout the duration of a strategic node. The multiplier of node n, M_n, is calculated as:

$$M_n = \frac{365}{|Seasons| * |Periods|} * (t(n+1) - t(n)) \tag{7.3}$$

where $|Seasons|$ is the number of seasons, $|Periods|$ is the number of periods in each scenario, and $t(n+1) - t(n)$ gives the duration of the strategic node n.

7.3.2 Lease Lengths and End of Horizon Effects

As mentioned in Chapter 5, lease lengths of vessels are represented by the year in which the lease expires. In order to ensure that the model can handle the fleet balance of chartered vessels correctly, these expiration years must correspond to the year of a strategic node in the scenario tree. As a result, the possible charter lengths get shorter and shorter downwards in the tree, and the set of possible charter lengths gets smaller. As chartering for a shorter time period is more expensive than chartering for many years, this results in increasingly expensive charter rates downwards in the tree. This might lead to solutions that rent too much in the early nodes to satisfy demand in later nodes. Such problems are often referred to as end of horizon effects, and are common when a finite and fixed horizon is imposed on a problem which may have a longer horizon in reality.

To counteract this issue, the possibility of chartering a vessel for at least 10 years in every node has been added. For nodes where the remaining duration of the planning horizon is less than 10 years, the total charter cost have been adjusted so that costs running outside the planning horizon are not taken into account. As an example, in the leaf nodes, that have a one year duration, this gives the wind farm owner the alternatives of either chartering vessels for 1 year at a high price, or chartering vessels for 10 years and pay 1/10 of the lower 10-year-charter cost. As a result, if any vessels are chosen to be long-term chartered in the leaf nodes, they will always be chartered for 10 years. Hence, alternative charter lengths have been removed in the root node.

7.3.3 Penalty Costs

In the mathematical model, penalty costs are incurred when the decision-maker decides not to conduct a certain maintenance task on a turbine. Finding an appropriate level for the penalty costs is important in order to encourage a balanced fleet size and mix. A penalty cost that is too low leads to no maintenance being conducted at all, as this makes it cheaper to pay the penalty cost than to acquire vessels to conduct maintenance. On the other hand, a penalty cost that is too high eliminates the effect of allowing maintenance tasks not to be conducted in the first place. The method for calculating the penalty costs used in the computational study is inspired by the previous work of Raknes et al. [116], and Gundegjerde and Halvorsen[64].

The penalty cost for preventive and corrective maintenance tasks are calculated differently. For preventive tasks, the consequence of not conducting a task is the increased probability of failures occurring in the future. To ensure that preventive tasks are given an incentive to be conducted, the penalty cost is calculated as the largest possible cost of conducting the task. The cost of conducting a preventive maintenance task, when considering the vessel acquisition as sunk, is the downtime cost incurred while maintenance is being conducted, and the variable cost of using a vessel to conduct it. The largest possible cost of conducting preventive maintenance would hence be incurred if the vessel with the highest variable cost is used at the time where the downtime cost is highest. The downtime cost is at its highest when the turbines at a wind farm produce at their maximum effect. The penalty cost of not completing a preventive task can hence be calculated as:

$$
\begin{aligned}
&\text{Preventive Penalty Cost [€]} = \text{Max Effect of Turbine [MW]} \\
&* \text{(Electricity Price + Subsidy)[€/MWh]} * \text{Man-Hours [h]} \\
&+ \text{Variable Cost of Vessel [€/h]} * \text{(Transit Time to Farm + Man-Hours) [h]}
\end{aligned} \tag{7.4}
$$

For corrective maintenance, the consequence of not conducting a maintenance task is revenue loss due to a turbine not being able to generate power. As this is a considerable cost, the penalty cost of not conducting a corrective task should be larger than for preventive tasks, to give an incentive to prioritize corrective maintenance over preventive maintenance. The penalty cost of not conducting a corrective task is set equal to the short-term charter rate of the cheapest vessel that is able to conduct the task. In this way, the cost of chartering an extra vessel to be able to conduct the task with certainty is equal to the penalty cost incurred if the fleet is unable to meet demand for maintenance.

7.3.4 Other Calculations

Downtime costs have been calculated as described in Subsection 2.6.1 and Subsection 5.1.7. The travel time for each vessel type to each wind farm is calculated based on the speed of the vessel type and the distance from the harbour or offshore station that the vessel type belongs to. Furthermore, the scenario generator calculates the total charter cost of vessels based on yearly charter rates and lease lengths. For a given lease length, the total charter cost, TC, is calculated as:

$$\text{TC [€]} = \text{Yearly Charter Cost [€]} * \text{Lease Length [years]} * (1 - \text{Discount}) \qquad (7.5)$$

The discount is conditional on lease length, and is included to reflect the fact that vessel agreements with long lease lengths have a lower yearly cost (as described in Section 2.4.4).

7.3.5 Preprocessing

To reduce the number of constraints in the problem, some simple preprocessing is conducted in the scenario generator. Constraints (5.13) are preprocessed and removed from the model by making the utility ratio E_v dependent on node, season, scenario and period. The new utility ratio E_{nqspv} is set to zero if the weather capabilities of a vessel type is exceeded by the weather realization in a given period.

7.4 Evaluating the Scenario Generation Method

In the generation of tactical scenarios, randomness is introduced by the scenario generator in two ways: by sampling weather conditions from historical weather data and by randomly generating demand for corrective maintenance. When stochastic parameters are discretely approximated with this approach, a concern about the quality of the scenario generation method is introduced. Kaut and Wallace introduce a method for testing the practical performance of a scenario generation method in [81]. They present two minimal requirements that scenario generation methods must satisfy: stability and bias. These requirements are important to ensure that the generated scenarios are usable for a given model, and to ensure a certain quality of the solution found by the optimization model. Furthermore, the paper separates between two types of stability: in-sample and out-of-sample stability [81].

According to Kaut and Wallance, stability is obtained if all scenario trees generated on the same input gives the same objective value and solution in the scenario-based optimization problem [81]. In-sample stability guarantees that when generating several scenario trees with the same input, the optimal value of the objective function and solution reported by the model itself is (approximately) the same, regardless of the scenario tree that is used [81]. Formally, if K scenario trees are generated with the discretisation $\{\breve{\xi}_{tk}\}$ for the stochastic process $\{\tilde{\xi}_t\}$, then the in-sample stability can be defined as:

$$F(x_k^*; \breve{\xi}_{tk}) \approx F(x_l^*; \breve{\xi}_{tl}) \qquad k, l \in 1, \ldots, K \qquad (7.6)$$

where x_k^* is the optimal solution found when solving the optimization problem with each of the scenario trees, $k = 1, \ldots, K$, and $F(x_k^*; \breve{\xi}_{tk})$ is the objective function obtained when the process $\{\tilde{\xi}_t\}$ is approximated by a scenario tree $\{\breve{\xi}_{tk}\}$.

Out-of-sample stability is obtained if the solutions of all scenario trees generated with the same input get (approximately) the same value as the value of the true objective function, $F(x; \tilde{\xi}_t)$ [81]. Out-of-sample stability is only necessary to test if in-sample stability in solution is not obtained, as it is otherwise guaranteed. However, the reverse does not hold. Using the same notation as described above, out-of-sample stability can be defined as:

$$F(x_k^*; \tilde{\xi}_t) \approx F(x_l^*; \tilde{\xi}_t) \qquad k, l \in 1, \ldots, K \tag{7.7}$$

If the scenario generation method has out-of-sample stability, the real performance of the solution found by the optimization model is stable [81]. The scenario generation method would hence provide solutions of good quality. However, if the scenario generation method does not have in-sample stability as well, it would not be obvious how good these solutions actually are. In order to evaluate how good the quality of the solution is, it would be then necessary to find an average solution value by solving several instances in the optimization model, as the different scenario trees might give different solutions. However, having out-of-sample stability without in-sample stability is still preferable to the opposite: having in-sample stability without out-of-sample stability. In the latter case, the real performance of the solution found would depend on the scenario tree used.

In a sampling method, the strongest candidate for instability is often the lack of scenarios [81]. The aim of stability testing can therefore be to investigate the number of scenarios needed to ensure stability in the scenario generation method. In-sample stability is tested by generating several scenario trees on the same input, and comparing the objective function value and solution found by the optimization model. Bias and out-of-sample stability is hard to test, as testing requires evaluation of the "true" objective function value, found by solving the problem with the "true" continuous distribution of the stochastic parameters[81]. Regardless, testing of the out-of-sample stability can be conducted by approximating the "true" distribution with a large reference tree. Bias has not been tested for the DLPOW, and is therefore not presented further.

7.5 Evaluating the Dual-Level Stochastic Model

Stochastic models are often computationally demanding, and require specific solution methods. It is therefore beneficial to evaluate whether using a stochastic model is necessary to begin with, or if a deterministic approach would be sufficient. Two common methods for evaluating stochastic models are the value of stochastic solution (VSS) and the expected value of perfect information (EVPI). EVPI is defined as the amount of money a decision-maker is willing to pay in return for complete information about the future [20]. Obtaining complete information on future electricity and vessel prices, introduction of new vessel concepts, and delays in stepwise wind farm development is impossible. EVPI can therefore be seen as a purely theoretical value for the model for DLPOW. According to Birge et al. [20], when this is the case and no future information is available, the VSS becomes more practically relevant. Hence, focus has been put on evaluating VSS in the computational study and EVPI is therefore not discussed further.

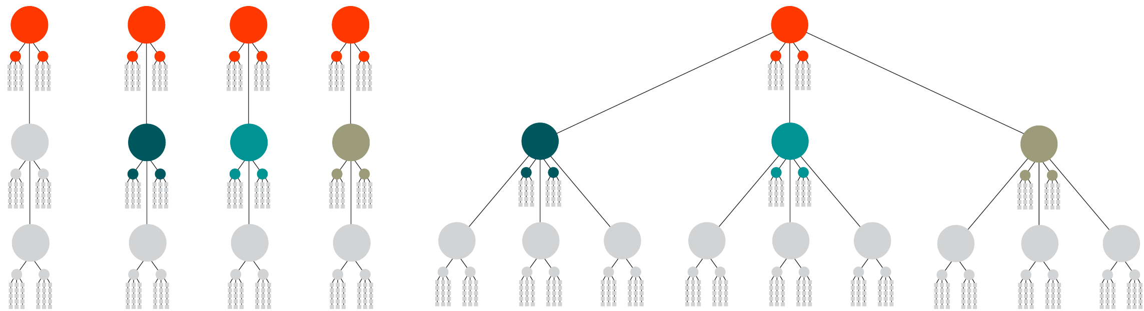
VSS measures the value of using a stochastic approach over a deterministic approach, and hence measures the value of knowing and using probability distributions on future outcomes [20]. The VSS is calculated as the difference between the expected value of using an expected value approach (EEV) and the optimal solution to the stochastic problem (SP). To find the EEV, the solution from solving the expected value (EV) problem is used. The EV problem is a deterministic version of the problem, where all stochastic parameters have been replaced with their expected values. For a traditional stochastic two-stage model, the EEV is found by solving the SP with the first stage decisions fixed to the solution found in the EV problem [20].

7.5.1 Value of Strategic Stochastic Solution

While VSS has been frequently used to evaluate traditional two-stage and multi-stage stochastic models, it has not, to the authors knowledge, been applied on dual-level stochastic models. Using VSS on a dual-level stochastic model is complicated in terms of deciding how and which variables to fix at strategic and tactical level when calculating the EEV. Our approach is to only consider the strategic nodes when calculating VSS, resulting in what we call the Value of the Strategic Stochastic Solution (VSSS). VSSS measures the value of introducing strategic uncertainty on top of a stochastic model with tactical uncertainty. In the VSSS calculation, only decisions made at the strategic level are fixed when calculating the EEV. The decisions made at tactical level are never fixed. The VSSS is calculated with a similar approach as used for calculating a traditional VSS for multi-stage stochastic models. However, in the VSSS calculation, the embedded tactical scenarios are included in all strategic nodes when finding the EV solution. Hence, our method differ from the traditional approach by including scenarios at tactical level in the otherwise deterministic problem used to find the EV solutions.

Calculating V(S)SS for multi-stage stochastic models is not straightforward, as (strategic) decisions are made on several stages. A decision regarding which (strategic) variables to fix at each stage must therefore be made. A trivial approach would be to only fix decisions made in the first stage. Such a method would, however, not be sufficiently beneficial to the stochastic model developed in this thesis, as the fleet size and mix can be modified in all strategic nodes at consecutive stages in the EEV after uncertain parameters are revealed. Another approach suggested in [20], is to determine the EV by solving the expected value problem for all stages and fixing these decisions in all stages when solving the EEV. Using this approach to calculate the VSSS for our model would enforce the same fleet size and mix in all strategic nodes at the same stage in the scenario tree, giving a large advantage to the stochastic model. In yet another approach, suggested in [54], a chain of EV problems are solved. With this approach the deterministic model used to find the EV solutions, is allowed to update decisions at the consecutive stages based on new information. This approach has been used when calculating VSSS in the computational study.

Figure 7.4 illustrates how the EEV is calculated for the DLPOW for a dual-level problem with 3 stages in the strategic scenario tree. Figure 7.4 (a) illustrates the EV1 problem that is solved to find the EV solution for the first strategic node of the tree. The EV2 problems, one for each strategic node at the second stage, find the optimal solution to the EV problem in the respective strategic node. In the EV2 problems, the strategic decisions in the first node have been fixed to the optimal solution of EV1. This is illustrated in Figure 7.4 (b). The EEV is then calculated by fixing the strategic decisions in the first node to the optimal solution found in the EV1 problem, and fixing the strategic decisions made in the nodes on the second stage to the solutions found in the respective EV2 problems. This is illustrated in Figure 7.4 (c). The number of EV2 problems is always equal to the number of strategic nodes at the second stage in the SP.



(a) EV1 (b) EV2 (c) EEV

Figure 7.4: Illustration of EEV calculation used to calculate VSSS.

When the EEV is calculated, the VSSS is given as:

$$VSSS = EEV - SP$$

For a minimization problem, Birge and Louveaux [20] show that the following property must hold for all stochastic problems:

$$EEV \geq SP$$

If the property does not hold, the SP is not the optimal solution to the stochastic problem, as the expected value solution also is valid for the stochastic problem, and hence could have been chosen to get a better solution [20].

Chapter 8

Computational Study

In this chapter, the mathematical model, scenario generator and solution methods for the DLPOW are tested by the use of computational experiments. In Section 8.1, the number of scenarios and periods needed to ensure in- and out-of-sample stability in the scenario generator is investigated. The solution methods used to solve the DLPOW are tested in Section 8.2 - Section 8.4. The performance of the Greedy Tactical Heuristic, used to solve tactical subproblems in the GRASP, is tested in Section 8.2. In Section 8.3, calibration testing is conducted on the GRASP, in order to find the best parameter settings. The performance of the GRASP and the commercial optimization solver is examined in Section 8.4. In Section 8.5, the value of accounting for uncertainty at both strategic and tactical level in one optimization model is investigated through conducting VSSS calculations.

In the computational experiments described throughout this chapter, several test cases are solved by the metaheuristic described in Chapter 6 and/or by the commercial optimization solver. In this master thesis, a test case is defined as a set of test instances generated on the exact same input. All input parameters in a test case are hence equal. A test instance is simply a problem instance, and two test instances within the same test case only differ in their tactical scenario tree. All test instances used in the computational study are generated based on the input presented in Section 7.1, unless otherwise specified.

All tests conducted in the computational study are performed on a computer with an Intel(R) Core(TM) i7-3770, CPU 3.4 GHz processor and 16 GB RAM. All tests instances solved with the commercial optimization solver, FICO(TM) XPress-IVE Version 1.24.06 64-bit, have been run with standard parameter settings and an aggressive cutting strategy. All test instances solved with the GRASP in Eclipse Mars 4.5.1 have been solved in the Java Runtime Environment(JRE) 1.8.0_73, with a heap-space limitation of 4 GB.

8.1 Stability in the Tactical Scenario Generator

As discussed in Section 7.4, stability in the scenario generator is desired to ensure that the solutions found by the optimization model can be trusted to be representative for the input provided, and do not depend on the specific problem instance (scenario tree) generated. The tactical scenario generator, presented in Subsection 7.2.1, is tested for in-and out-of-sample stability in this computational study. Stability testing is not conducted for the strategic scenarios, as these are not generated randomly and such testing is hence not necessary. The aim of the stability testing is to investigate the number of tactical scenarios needed to ensure stability, and to investigate how the number of periods in each tactical scenario affect stability.

As the stability testing is conducted to test stability in the generation of tactical scenarios, the test instances that have been used only have one strategic node with a duration of one year. Furthermore, all test instances have 2 farms, 1 harbour, 1 offshore station, and 4 vessel types. All test instances have been solved in the optimization solver. The following subsections present the results from the stability testing.

8.1.1 In-Sample Stability

In order to test the scenario generation method for in-sample stability, 28 test cases with 15 test instances each, have been solved. The test cases only differ in the number of tactical scenarios and periods, ranging from 5 - 120 scenarios and 3 - 15 periods. Each test instance has been solved to an optimality gap of 0.05%. In order to evaluate the in-sample stability, the coefficient of variation (COV) between the 15 test instances within each test case has been used as a measure of spread in objective function value. The COV expresses the amount of variability relative to the average objective function value, and has for each test case been calculated as:

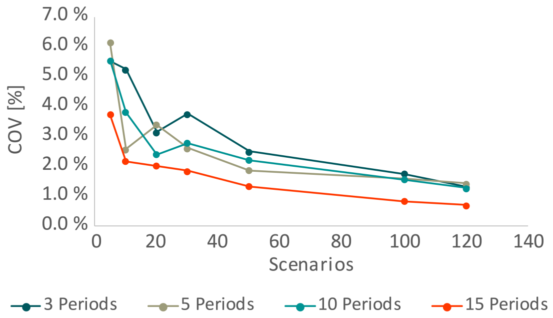
$$COV = \frac{\sigma}{\mu} \tag{8.1}$$

where σ is the standard deviation, and μ is the average objective function value in the test case.

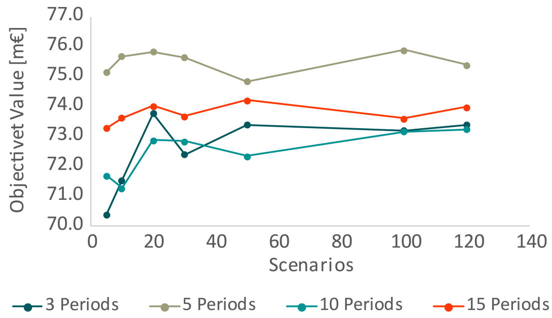
Table 8.1, Figure 8.1 and Figure 8.2 show the results from the in-sample stability testing. Table 8.1 presents the COV in objective function value for each test case. Figure 8.1 (a) shows a graphical representation of the COV for each test case, while Figure 8.1 (b) illustrates the average objective function value for each test case. Finally, Figure 8.2 shows the average computational time needed to solve one instance from each test case.

Table 8.1: COV in the in-sample stability test cases.

Scenarios	3 Periods [%]	5 Periods [%]	10 Periods [%]	15 Periods [%]
5	5.52	6.14	5.56	3.73
10	5.25	2.58	3.82	2.17
20	3.14	3.39	2.41	2.03
30	3.75	2.61	2.79	1.85
50	2.50	1.88	2.22	1.34
100	1.76	1.58	1.56	0.85
120	1.32	1.44	1.28	0.72



(a) COV in objective function value for each test case.



(b) Average objective function value of each test case.

Figure 8.1: Results from in-sample stability testing.

As can be seen in Table 8.1 and Figure 8.1 (a), the in-sample stability clearly increase with the number of tactical scenarios. This is expected, as the impact of extreme value scenarios on the solution generally decrease with an increasing amount of scenarios. All test instances from 100 scenarios onwards have a COV lower than 2%, regardless of the number of periods. The lowest COV of 0.72% is found for the test case with 120 scenarios and 15 periods. Additionally, Figure 8.1 (b) show that the average objective function values are highly unstable for all test cases with less than 50 scenarios. Based on these results, 100 scenarios is considered as an absolute minimum for ensuring in-sample stability for 3, 5, and 10 periods, while 50 scenarios is seen as a minimum for 15 periods.

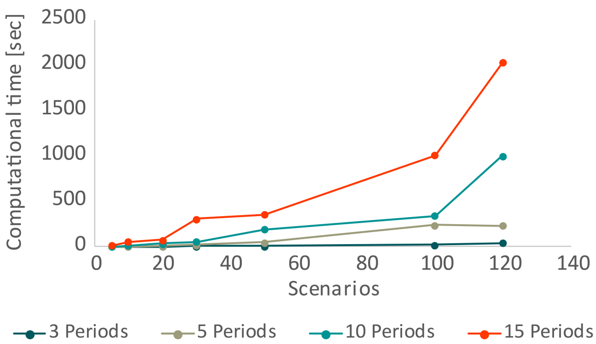


Figure 8.2: Average computational time for each test case.

Considering the effect of periods on the in-sample stability, it can be seen from Table 8.1 and Figure 8.1 that using 15 periods gives the best COV for all scenarios, and that an increased number of periods generally seems to give a better in-sample stability. The results reported in Figure 8.1 (b) show that the average objective function value reported by test cases with 5 periods generally are too high, compared to the other test instances. The reason for this disparity is not evident, but 5 periods is consequently considered an unsuitable choice to ensure stability. Based on the results presented above, the choice of 10 or 15 periods appear best in order to ensure in-sample stability. However, when considering the computational time, the results in Figure 8.2 show that the number of periods clearly affect computational time. Comparing test cases with the same amount of scenarios, the average computational time for a test instance with 15 periods is significantly higher than for a test instance with a lower number of periods. Hence, when deciding the number of periods, a trade-off between computational time and in-sample stability must be made.

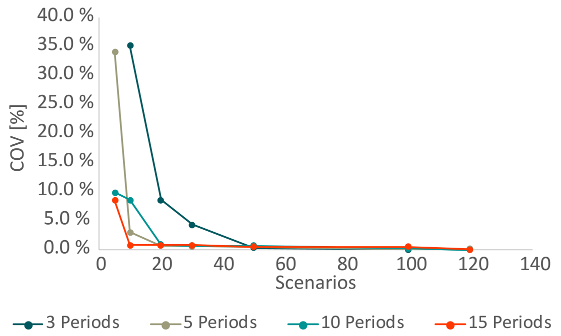
The results discussed so far measure the *in-sample stability in objective function value.* However, it is possible to have *in-sample stability in solution* without having in-sample stability in objective function value. For this reason, the fleet size and mix solutions found in each test instance of all test cases have been inspected, where the aim of the inspection is to asses the degree of in-sample stability in the solutions found. Looking at the various solutions found, it is evident that the number of different solutions found within each test case decrease with an increasing number of scenarios. For all test cases with 120 scenarios, only 2 different solutions are found. For the test cases with 5 scenarios up to 10 different solutions are found. This indicates a higher degree of in-sample stability in solution for test cases with a high number of scenarios. However, no test case in the tested range reach perfect in-sample stability in solution, which require only one solution to be found across all 15 test instances. In order to conclude on the number of tactical scenarios and periods required to ensure stability, out-of-sample stability testing is therefore necessary.

8.1.2 Out-Of-Sample Stability

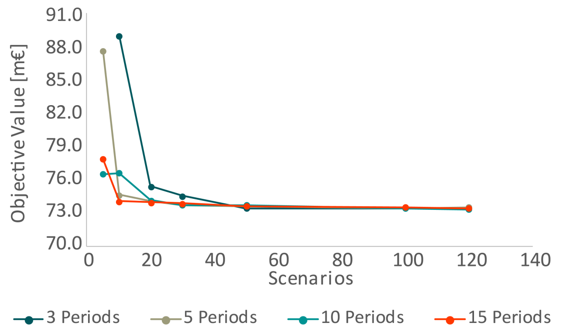
In the out-of-sample stability testing, the real distribution of uncertain parameters have been approximated through a reference scenario tree consisting of 10,000 scenarios and 10 periods in each tactical scenario. All the fleet size and mix solutions found in the in-sample stability testing have been tested on the reference tree to evaluate the real performance of these solutions. The real performance is given by the objective function value obtained when solving the reference tree, with the strategic solution fixed to the values found in a test instance. For each of the 28 in-sample test cases, the solutions found for all 15 test instances within the respective test case are used to calculate an out-of-sample COV, as a measure of the stability in real performance. Table 8.2 and Figure 8.3 show the results from the out-of-sample stability testing. Table 8.2 presents the COV in the objective function values obtained from the reference tree for the different solutions found in each test case. Figure 8.3 (a) shows a graphical representation of the COV for each test case, while Figure 8.3 (b) illustrates the average objective function value found by the reference tree for each test case.

Table 8.2: COV in the out-of-sample stability test cases.

Scenarios	3 Periods [%]	5 Periods [%]	10 Periods [%]	15 Periods [%]
5	-	34.05	9.87	8.54
10	35.19	3.07	8.65	0.82
20	8.62	0.71	0.91	0.82
30	4.35	0.61	0.75	0.80
50	0.41	0.53	0.77	0.61
100	0.06	0.22	0.31	0.56
120	0.06	0.23	0.04	0.07



(a) COV in objective function value for each test case.



(b) Average objective function value of each test case.

Figure 8.3: Results from out-of-sample stability testing.

The results from the out-of-sample stability testing show that the out-of-sample stability improves for an increasing number of scenarios. For all test cases with more than 50 scenarios, the COV is below 1% regardless of the number of periods in the tactical scenarios. The lowest COV in the tested range is found for the test case with 120 scenarios and 10 periods. The results from the reference tree indicate that, even though perfect in-sample stability is not reached for any number of tactical scenarios, the different solutions found for instances with more than 50 scenarios are approximately identical in real performance. These test cases can hence be regarded as out-of-sample stable. Furthermore, the results show that for a lower number of scenarios, the test cases with a lower number of periods perform significantly worse. However, for test cases with 50 scenarios or more, the number of periods does not appear to influence the stability to any large extent. Figure 8.3 (b) show that the average objective value stabilise for test instance with more than 50 scenarios, and that the value converges towards the same value for all test instances, regardless of the number of periods. Furthermore, Figure 8.3 (b) shows that the performance of a solution (given by the objective value of the solution found by the reference tree) improves as the number of scenarios increases. Based on these results, 50 scenarios is seen as the absolute minimum for ensuring out-of-sample stability for all periods.

8.1.3 Conclusion of Stability Testing

As the results from the in-sample testing indicate, perfect in-sample stability in solution and objective value is not obtained within the tested range. However, for the DLPOW, the strategic here-and-now decisions are considered most important. It is hence considered sufficient to have a level of stability where the real performance of the different strategic fleet size and mix solutions found is approximately the same. When choosing the number of periods and scenarios, ensuring the highest possible degree of out-of-sample stability has therefore been emphasised. The slight in-sample instability for test cases with more than 50 scenarios is not considered problematic, as these test cases are considered as out-of-sample stable, and hence give strategic solutions with approximately the same real performance.

Comparing the average objective function values reported in the in- and out-of-sample-testing, an observation can be made: for many test cases with few scenarios, the in-sample objective values reported are significantly lower than the out-of-sample (true) values. According to Kaut et al. [81], this means that the real performance of these solutions are notably worse than what the model states (the solutions have a higher 'true' objective function value). This is a common observation, and is caused by the fact that when the number of scenarios is low, the solution found is over-fitted to each scenario, and hence the model underestimates the 'real cost' [81]. From this, it can be seen that the test cases with 15 periods never underestimate costs. However, all test cases with 10 periods and less than 100 scenarios underestimate costs. In light of this observation, the number of periods and scenarios should be chosen in such a manner that the objective function value reported by the model is approximately equal to the 'true' objective function value, reported by the out-of-sample reference tree.

On account of the discussion above, the choice of 120 tactical scenarios and 10 periods is considered as the best option. This is due to the fact that this test case have the lowest out-of-sample COV, a relatively low in-sample COV, and an average in-sample objective function value that coincides with the value that the average out-of sample objective converge towards (73 m€). The test case with 100 scenarios and 3 periods could also be a good option, as this test case only has a slightly worse out-of-sample COV, report approximately the true objective function value, and has a significantly lower average computational time. However, when also taking in-sample stability into consideration, 120 scenarios and 10 periods performs better than 100 scenarios and 3 periods. As ensuring stability is considered more important than having low computational time, 120 scenarios and 10 periods is regarded as the best option. In the remaining part of the computational study, all test instances used have been generated with 120 scenarios and 10 periods (unless otherwise specified).

8.2 Performance of the Greedy Tactical Heuristic

In this section, the performance of the Greedy Tactical Heuristic used to solve tactical subproblems, presented in Subsection 6.5, is tested. The performance of this heuristic is crucial for the overall performance of the GRASP, as its results are used to evaluate candidate insertions when constructing solutions. As the optimization solver has perfect look-ahead and utilize more information when making decisions, it is unrealistic to expect a simple greedy heuristic to make the exact same choices as the optimization solver. In Subsection 8.2.1, a simple test by inspection is therefore conducted in order to identify similarities and differences between the tactical decisions made by the heuristic and the optimization solver. In Subsection 8.2.2, the difference in tactical cost resulting from unsimilar decisions made by the two solution methods is evaluated. Furthermore, when considering the role of the Greedy Tactical Heuristic in the GRASP, the most important concern is that the Greedy Tactical Heuristic deploys the fleet in such a way that "good" fleet size and mix solutions result in lower costs than "bad" fleet size and mix solutions. This is tested in Section 8.2.3.

8.2.1 Differences in Tactical Deployment Decisions

As mentioned, the optimization solver and the Greedy Tactical Heuristic use fundamentally different approaches to solve the DLPOW. As the solution methods use different logic when making deployment decisions, the Greedy Tactical Heuristic is expected to make somewhat different tactical decisions. To identify and analyse these differences, some simple test instances have been solved with the strategic solutions fixed to a given fleet. The tactical decisions made by the two solution methods have subsequently been compared to each other by inspection. The differences in logic observed, and their effect on tactical decisions and costs, are discussed in the following paragraphs.

Difference in Utilization of Vessels in Periods

In order to keep the downtime costs of corrective tasks low, the Greedy Tactical Heuristic attempts to conduct corrective tasks as soon as possible, by exhausting the fleet (or conducting all uncompleted tasks) in each period. In each period, the Greedy Tactical Heuristic chooses to exhaust the capacity of the cheapest vessel first, before considering the second cheapest vessel. However, the heuristic does not consider that it might be cheaper to wait until a later period to conduct a corrective maintenance task with a cheaper vessel, rather than doing it as soon as possible with a more expensive vessel. This results in an over utilization of expensive vessels in the Greedy Tactical Heuristic compared to the optimization solver. In turn, this results in increased variable cost, and decreased corrective downtime cost in the Greedy Tactical Heuristic.

When conducting preventive maintenance, the Greedy Tactical Heuristic attempts to utilize the cheapest vessel in all periods before it considers using the second cheapest vessel in any period. The periods are sorted in a greedy manner, and the heuristic first conducts as much preventive maintenance as possible with the cheapest vessel in the period with the lowest preventive downtime cost. However, the heuristic does not consider the trade-off between using an expensive vessel in a cheap period versus using a cheap vessel in an expensive period. The Greedy Tactical Heuristic therefore generally decides to conduct more preventive maintenance in more expensive periods compared to the optimization solver. This results in higher preventive downtime costs in the Greedy Tactical Heuristic. However, when conducting preventive maintenance, the heuristic utilize cheaper vessels to a larger extent than the optimization solver, resulting in decreased variable costs.

Difference in Utilization of Vessels at Farms

While the optimization solver can consider all possible combinations of sending different vessels to different farms in various periods, the Greedy Tactical Heuristic only considers one period and one vessel type at the time. The heuristic therefore struggles more with utilizing the capacity of the fleet in an optimal way, when considering the amount of vessels to send to each farm in each period. Generally, the heuristic tends to send out marginally more vessels compared to the optimization solver, which leads to higher transit costs in the heuristic.

When conducting preventive maintenance, the Greedy Tactical Heuristic only considers one vessel type and one period at the time. When sending out vessels from depot to farms, the heuristic do not take into consideration that there might already be a vessel with spare capacity located at the farm in the same period, or that a vessel sent to the farm in another period has spare capacity. Hence, in some cases, the heuristic sends too many vessels to one farm, rather than exploiting the spare capacity of vessels already located at the farm. Overall, this may result in spare capacity at one farm, and lack of capacity to finish all preventive tasks at another farm. In result, the heuristic sometimes faces penalty costs that the optimization solver avoids.

Furthermore, the Greedy Tactical Heuristic does not take into consideration the fact that if it is not possible to finish a preventive task, it is better to not start the task at all. Hence, the heuristic might choose to do a fraction of one task. This is a choice that the optimization solver never makes, as it leads to an increase in variable costs without any decrease in penalty costs. Furthermore, when conducting maintenance, the Greedy Tactical Heuristic selects a farm based on a myopic criteria every time one vessel have been exhausted. As the farm with the highest cumulative cost is chosen first, this does in some cases result in using the last capacity of the fleet to do a fraction of a task at one farm, rather than completing a task started at another farm. These differences in decision-making result in higher penalty costs in the heuristic. This is especially the case when evaluating fleets with under-capacity or with little slack in terms of available operational hours.

8.2.2 Differences in Tactical Costs

As the Greedy Tactical Heuristic does not always make the same tactical decisions as the optimization solver, it is important to evaluate how the difference in decisions affects the total tactical cost. This has been tested by comparing the tactical costs found by the Greedy Tactical Heuristic against the tactical costs found by the optimization solver for a set of test instances.

4 test cases, each with 10 test instances, have been evaluated. The test cases all have 1 strategic node and 120 scenarios, and the test instances in the 4 test cases only differ in the amount of periods used in each tactical scenario. In addition to evaluating the difference in tactical costs, the test cases are also used to evaluate whether the number of periods affect the difference in cost. The fleet size and mix solution has been fixed in all test instances at: $x_{1,3,10} = 2$, $x_{1,4,10} = 2$, $y^{IN}_{1,1,3} = 2$, resulting in a fleet of 6 vessels during summer, and 4 vessels during winter. This fleet size and mix solution has been chosen as it occurs most often in the most stable in-sample test cases presented in Section 8.1. To evaluate the differences in tactical costs between the two solution methods, a percentage deviation is calculated. For each test instance, the percentage difference between the tactical cost found by the Greedy Tactical Heuristic and the tactical cost found by the optimization solver is given by:

$$\text{Difference [\%]} = \frac{\text{Tactical Cost [Heuristic] - Tactical Cost [Solver]}}{\text{Tactical Cost [Solver]}} * 100 \tag{8.2}$$

A positive percentage indicates that the Greedy Tactical Heuristic gives a higher (worse) tactical cost than the optimization solver. The results from the testing are shown in Figure 8.4, which reports the average difference of the 10 test instances in each test case. As the figure shows, the average tactical cost found for a given fleet is 1.5-3.0% higher in the Greedy Tactical Heuristic than in the optimization solver. The average difference in tactical cost is higher during winter than during summer, regardless of the number of periods. Furthermore, Figure 8.4 shows that the average difference in total tactical cost is higher for the test case with 3 periods than for any other test case.

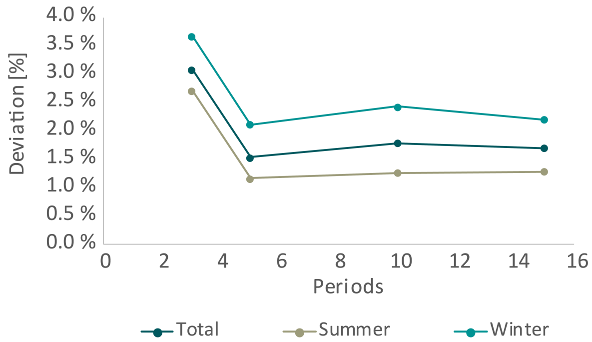


Figure 8.4: Average differences in tactical costs.

The difference in tactical costs between the two solution methods indicates that the Greedy Tactical Heuristic is not able to utilize the given fleet as well as the optimization solver, especially in the winter season. In winter, the weather conditions are harsher and the time windows for maintenance are hence shorter. This increases the importance of optimal fleet deployment, and the consequence of not being able to deploy the fleet optimally hence becomes more severe. As discussed in the previous section, the Greedy Tactical Heuristic does not deploy the fleet optimally, resulting in higher tactical costs when using this solution method. However, it can be argued that the decisions made by the Greedy Tactical Heuristic are more realistic than the decisions made by the solver. The optimization solver has perfect look-ahead, and for this reason it is likely to over-optimize. The over-optimization makes it hard for a real-life wind farm owner to actually accomplish the suggested tactical deployment of a fleet from the solver. In result, the optimization solver might tend to choose a fleet size and mix solution that is too small, compared to what would be optimal in real-life. This kind of over-optimization is not performed by the Greedy Tactical Heuristic, and the tactical costs and decision made in the heuristic can hence be regarded more realistic.

8.2.3 Evaluation of Fleet Size and Mix Solutions

As shown in the previous subsections, there is a slight difference in both decisions and tactical costs reported by the two solution methods. This calls for further evaluation to ensure that these differences do not influence what is considered as the best strategic fleet size and mix solution. The aim of the testing in this subsection is hence to analyse whether both solution methods consider the same fleet size and mix decisions to be best or worst. For this to be the case, a solution that is considered to be good (low objective value) in the optimization solver also needs to be considered good by the Greedy Tactical Heuristic, and vice versa.

In order to test how different solutions are evaluated, 10 different fleet size and mix solutions (fleets) have been fixed in the same test instance. For each fixed fleet, the test instance has been solved once by each solution method. The test instance used have 1 strategic node, 120 tactical scenarios and 10 periods. Table 8.3 shows the objective function value found by each solution method for the various fixed fleets. Furthermore, the table shows how the different fleets are ranked within each solution method, based on their objective function value. The last column gives the difference in the tactical costs found by the two solution methods, calculated as described by Equation 8.2. An overview of the 10 fleet size and mix solutions used is given in Appendix B.2.

Table 8.3: Ranking of fleets according to objective function value.

	Optimization Solver		Greedy Tactical Heuristic		
Fleet	Objective [m€]	Rank [-]	Objective [m€]	Rank [-]	Difference [%]
1	73.77	1	74.32	1	1.70
2	74.02	2	74.37	2	1.13
3	74.11	3	74.54	3	1.36
4	75.60	4	76.86	5	3.65
5	75.64	5	76.90	6	3.57
6	76.43	6	76.84	4	1.33
7	82.10	7	83.56	7	3.55
8	114.89	8	117.78	8	3.76
9	201.22	9	207.26	9	3.65
10	1,550.64	10	1,550.64	10	0.00

The results from Table 8.3 show that both the optimization solver and the greedy heuristic rank the best and the worst fleet size and mix solutions equally. 7 of the 10 fleets have the same rank in both solution methods, where the best fleets (1, 2, and 3) and the worst fleets (7, 8, 9, and 10) are ranked equally. However, the fleets ranked in the middle (4, 5, and 6), are ranked differently. This could be due the fact that these fleets are very close in terms of objective function value in both solution methods, with an average percentage difference of 0.73% in the objective function values found by the optimization solver. However, the average percentage difference in the objective function values found by the solver for fleets 1, 2 and 3 is 0.31%. This shows that the heuristic is able to differentiate solutions that are almost equal in terms of objective function value, and the different rankings of fleets 4, 5 and 6 can hence not be explained solely by the closeness in objective function value.

The dissimilar ranking of 4, 5, and 6 can more likely be explained by the differences in fleet size between these instances. As the Greedy Tactical Heuristic struggle with deploying a fleet optimally, it may prefer fleets that are slightly larger than what the optimization solver prefers. Furthermore, an inspection of the results shows that the difference in the tactical cost found by the two solution methods are generally larger for small fleets, than for large fleets. This results in a slight rearrangement in the ranking performed by the heuristic compared to the optimization solver. However, as these fleet size and mix solutions are approximately equal in terms of objective function value, and the best fleets are still evaluated as best by the Greedy Tactical Heuristic, this slight rearrangement is not considered to be a major issue.

8.2.4 Conclusion of Performance Testing

As discussed in this section, there are differences in how the two solution methods make choices regarding deployment of a given fleet. Due to the lack of look-ahead in the Greedy Tactical Heuristic, this solution method is not able to utilize a given fleet as optimally as the optimization solver. This results in a small difference in tactical costs found by the Greedy Tactical Heuristic and the optimization solver, where the heuristic always returns a slightly higher cost of deploying a given fleet. However, despite these differences in deployment decisions and costs, it has been shown that the overall performance of the Greedy Tactical Heuristic is good. Both solution methods rank fleet size and mix solutions equally, and a good fleet in the optimization solver is likely to be evaluated as a good fleet in the heuristic as well. Hence, the performance of the embedded Greedy Tactical Heuristic is considered sufficiently good for its intended use in the GRASP.

8.3 Calibration of the Reactive GRASP

When using the reactive GRASP to solve the DLPOW, several parameters affect the performance of the metaheuristic. As presented in Subsection 6.2, the GRASP is executed with several input parameters: `Max_Iterations`, `Top_Down`, `Rank_Based`, δ, $A = \{\alpha_1, ..., \alpha_m\}$ and `Block_Iterations`. These parameters affect both the computational time and the quality of the solutions found by the GRASP. The aim of the calibration testing conducted in this section, is to examine combinations of parameter values and find the combination for which the GRASP performs the best. Hvattum et al. [74] state that due to the vast number of possible combinations of parameter values, exhaustive calibration testing of all possible combinations is not possible. Hvattum et al. therefore suggest that calibration testing can be conducted by examining the parameters in a sequential manner, where one parameter is changed at the time and untested parameters are kept constant at reasonable values.

To calibrate the GRASP, 10 test instances have been used. To keep the calibration testing unbiased towards a specific type of test instance, the 10 instances have been generated with different user input. The test instances vary greatly in size, having between 1 - 13 strategic nodes, arranged in strategic scenario trees of different shapes. Different strategic uncertainty is considered in the test instances, and each test instance accounts for either: no strategic uncertainty (Deterministic), uncertainty in electricity prices (Electricity Price) or uncertainty in stepwise development of wind farms (Farms). The number of vessel types vary from 4 - 6, and the number of wind farms vary from 2 - 3. All test instances have 120 scenarios and 10 periods. Table 8.4 contains a summary of the test instances, showing which type of strategic uncertainty each instance accounts for, the number of nodes at each level of the strategic scenario tree, the total number of strategic nodes, the number of vessel types, and the number of wind farms.

As discussed in Section 6.6, two versions of the reactive GRASP have been implemented: an any-node reactive GRASP (ANG) and a top-down reactive GRASP (TDG). These two versions differ in the way candidate insertions and solutions are constructed, and what is considered to be the the best parameter values may differ for these two versions. For this reason, the parameters are tuned separately for the ANG and for the TDG in the cali-

Table 8.4: Test instances used in the calibration testing of the GRASP.

Instance Name	Strategic Uncertainty	#Nodes at Each Stage	#Strategic Nodes	#Vessel Types (#Stations)	#Farms (#Turbines)
T1	Deterministic	(1, 1, 1)	3	4 (2)	2 (200)
T2	Deterministic	(1, 1, 1, 1, 1)	5	4 (2)	2 (200)
T3	Farms	(1, 2, 3)*	6	4 (2)	3 (300)
T4	Farms	(1, 3, 6)*	10	4 (2)	3 (300)
T5	Electricity Price	(1, 2)	3	4 (2)	2 (200)
T6	Electricity Price	(1, 2, 4)	7	6 (3)	2 (200)
T7	Deterministic	(1, 1, 1)	3	6 (3)	2 (200)
T8	Farms	(1, 3, 6)*	10	6 (3)	3 (300)
T9	Electricity Price	(1, 3)	4	6 (3)	2 (200)
T10	Electricity Price	(1, 3, 9)	13	6 (3)	2 (200)

*The strategic scenario trees for these instances are unbalanced. Their structure is explained in Subsection 7.2.2.

bration testing. Systematic testing has been conducted on the parameters `Rank_Based`, $A = \{\alpha_1, ..., \alpha_m\}$, and `Max_Iterations`, introduced in Chapter 6. The parameter δ, is set to 10 as suggested by Prais et al. in [111]. The parameter `Block_Iterations` is also set to 10. The calibration testing is conducted in three stages where different parameters are tuned in each stage. The tests conducted in the three stages are presented in Subsection 8.3.1 - Subsection 8.3.3, respectively.

8.3.1 Stage 1: Rank versus Value Based Selection

In stage 1, the aim is to analyse how the performance of the GRASP is affected by the use of a rank based versus a value based selection when constructing the RCL. The testing is done by running each of the 10 test instances twice in both the TDG and the ANG, once with a rank based selection and once with a value based selection. For the testing in this stage, the GRASP is run with `Max_Iterations` $= 1000$, and α-values are drawn from the set $A = \{0.00, 0.10, 0.20, ..., 1.00\}$. The test results from this stage are used to decide whether to use a rank based or value based selection in all further tests in the computational study.

The objective function values found for the test instances by the TDG and the ANG are shown in Table 8.5 and Table 8.6, respectively. The tables also depict the iteration number at which the best solution was found (*Best Itr*), the alpha value used when the best solution was found (*Best* α) and the computational time for 1000 iterations (*Total Time*).

The results reported in Table 8.5 show that the TDG finds solutions of equally good quality for 8 of the 10 tested instances, regardless of whether a rank based or a value based selection is used. However, for some of the larger instances (T4 and T8), the use of a value based selection results in a better solution. Furthermore, the results show that the use of a value based selection outperforms the use of a rank based selection in terms of computational time for 8 of 10 test instances. When a value based selection is used, the best solution is found at an earlier iteration for 9 of 10 instances. This could indicate that a lower number of iterations is needed if a value based selection is used.

Table 8.5: Results from stage 1 calibration testing of the TDG.

	Rank Based				Value Based			
Test Instance	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]
T1	885.06	44	0.10	278	885.06	4	0.00	100
T2	873.73	18	0.10	408	873.73	3	0.00	272
T3	1,039.47	867	0.10	1,352	1,039.47	16	0.00	790
T4	1,071.08	931	0.10	846	1,068.53	119	0.20	790
T5	913.62	23	0.10	232	913.62	1	0.00	215
T6	837.96	926	0.10	2,110	837.96	1	0.00	721
T7	874.06	104	0.10	605	874.06	5	0.00	362
T8	1,049.21	941	0.10	3,665	1,030.24	18	0.00	4,535
T9	893.09	61	0.10	509	893.09	17	0.00	362
T10	849.82	56	0.10	2,085	849.82	364	0.20	2,467

Table 8.6: Results from stage 1 calibration testing of the ANG.

	Rank Based				Value Based			
Test Instance	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]
T1	885.06	611	0.10	141	885.06	1	0.00	107
T2	873.67	428	0.20	319	868.28	6	0.10	213
T3	1,051.51	853	0.10	1,886	1,035.20	6	0.00	488
T4	1,110.11	28	0.10	1,689	1,068.31	171	0.10	1,760
T5	913.62	664	0.20	329	910.07	40	0.10	207
T6	843.89	377	0.10	1,397	837.96	10	0.00	902
T7	876.87	643	0.10	500	874.06	3	0.00	344
T8	1,082.45	487	0.10	5,346	1,028.65	36	0.00	4,593
T9	893.09	579	0.10	613	889.54	117	0.10	418
T10	873.98	169	0.10	4,631	849.28	650	0.20	2,718

The results for the ANG reported in Table 8.6 show that, similarly as for the TDG, the value based selection outperforms the rank based selection in terms of computational time and in terms of the iteration number at which the best solution is found. Furthermore, the table shows that a better solution is found for 9 of 10 instances when a value based selection is used in the ANG. Only for T1, the smallest test instance, does the solution quality not differ between the two approaches. Hence, the choice between a value based versus a rank based selection seems to have a higher consequence on solution quality for the ANG compared to the TDG. In conclusion, a value based selection is considered best for both the ANG and TDG, and is hence used in the remaining tests presented in this chapter.

8.3.2 Stage 2: Set of RCL Parameter Values $A = \{\alpha_1, ..., \alpha_m\}$

In the second stage of the calibration, the reactive extension of the GRASP is tested to find an appropriate set of α-values to be used in the construction of the RCL. The aim of the testing is to find a good maximum value for α, and the appropriate number of equidistant values to select from. Each test instance is solved by both the TGD and the ANG. The maximum value for α and the equidistance between the values are varied, while holding the minimum value of α constant at 0.00. As shown in the test results from stage 1, presented in Table 8.5 and Table 8.6, the best solutions in stage 1 were found for α-values in the range [0.00, 0.20] for both TDG and ANG, when a value based selection was used. Based on these results, both versions of the GRASP are tested with a maximum α-value of 0.20, to assess whether this leads to an improvement in performance. In order to evaluate how the number of values in the range affect performance, two different sets have been tested: $A1 = \{0.00, 0.10, 0.20\}$ and $A2 = \{0.00, 0.05, 0.10, 0.15, 0.20\}$, with the equidistances 0.1 and 0.05, respectively. The results from the testing are shown in Table 8.7 and Table 8.8.

Table 8.7: Results from stage 2 calibration testing of the TDG.

	$A1 = \{0.00, 0.10, 0.20\}$				$A2 = \{0.00, 0.05, ..., 0.20\}$			
Test Instance	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]
T1	885.06	1	0.00	41	885.06	10	0.00	30
T2	873.73	3	0.00	76	873.73	2	0.15	75
T3	1,039.47	3	0.00	184	1,039.47	3	0.00	177
T4	1,068.53	45	0.20	307	1,068.53	28	0.20	314
T5	913.62	1	0.00	40	913.62	1	0.15	40
T6	837.96	3	0.00	227	837.96	6	0.00	229
T7	874.06	3	0.00	96	874.06	1	0.10	88
T8	1,030.24	3	0.00	919	1,030.24	1	0.00	929
T9	893.09	4	0.00	149	893.09	7	0.00	133
T10	849.82	228	0.20	487	849.82	206	0.20	486

Table 8.8: Results from stage 2 calibration testing of the ANG.

	$A1 = \{0.00, 0.10, 0.20\}$				$A2 = \{0.00, 0.05, ..., 0.20\}$			
Test Instance	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]	Best Objective [m€]	Best Itr [#]	Best α [-]	Total Time [sec]
T1	885.06	2	0.00	52	885.06	6	0.10	48
T2	868.28	8	0.10	103	868.28	14	0.10	115
T3	1,035.20	3	0.00	259	1,035.20	4	0.00	260
T4	1,068.31	18	0.10	752	1,068.31	15	0.05	753
T5	910.07	7	0.20	51	910.07	12	0.05	49
T6	837.96	1	0.00	390	837.96	1	0.05	392
T7	874.06	2	0.00	121	874.06	5	0.05	124
T8	1,028.65	1	0.00	2,344	1,028.65	3	0.00	1,997
T9	889.54	115	0.10	189	889.54	8	0.05	187
T10	849.82	204	0.20	1,499	849.28	7	0.20	1,517

As the results show, reducing the maximum value on alpha from 1.00 to 0.20 does not deteriorate the quality of the solutions found by either the TDG or the ANG. For both GRASP versions, the same solution is found when using the sets $A1$ and $A2$ as with the original range used in stage 1 ($A = \{0.00, 0.10, ..., 1.00\}$). However, using the sets $A1$ and $A2$ results in a significant decrease in computational time for all test instances. Additionally, when 0.2 is used as the maximum value for α, 14 of 20 solutions found by the TDG and 15 of 20 solutions found by the ANG are found at an earlier iteration, when compared to using a maximum α-value of 1.0 as in stage 1. Furthermore, the results show that when considering computational time, solution quality, and number of iterations, $A1$

and $A2$ have no significant difference in performance. The results hence indicate that the number of equidistant values within the tested range does not affect the performance of the GRASP to any large extent.

Furthermore, as can be seen from the results from both stage 1 and stage 2, the same solutions are sometimes found for different α-values by both TDG and ANG. This shows that the GRASP is robust when it comes to parameter settings, and that the best solution found is not sensitive to specific parameter values. In conclusion, the results indicate that using a maximum value of 0.20 for alpha is desirable, as the computational effort is reduced without loss in solution quality. The set $A2$ has been chosen for use in further testing, as this set gives a slight increase in flexibility and diversification in the search, without any extra overhead in computational time.

8.3.3 Stage 3: Maximum Number of Iterations

The parameter `Max_Iterations` is a stopping criterion for the GRASP. Setting this parameter to a high value increases the probability of finding many different solutions, and hence the probability of finding the optimal solution [117]. Decreasing the number of iterations reduce this probability, but also the computational time. The number of iterations needed to ensure high-quality solutions depends on the parameter α . For a high α-value, where the GRASP is more random than greedy, the number of iterations needs to be high to ensure that most elements in the RCL are eventually investigated. However, if the α-value is zero, the RCL is smaller, and fewer iterations are needed. With a Reactive GRASP, where α is drawn randomly from a given range, the amount of iterations needs to be large enough for good α-values to be discovered and used.

The aim of the third stage of the calibration testing is to find an appropriate maximum number of iterations for the GRASP. As the test instances differ significantly in terms of complexity and size, the number of GRASP iterations needed to ensure high-quality solutions might differ between the instances. The number of strategic nodes and vessel types in a problem instance are especially expected to affect the number of iterations needed, as these directly affect the number of valid candidate solutions in each iteration. Rather than deciding one constant value of maximum iterations to be used for all test instances, the value of max iterations has been set relative to the size of the problem. This allows larger test instances to be run with a higher number of iterations. The number of iterations is considered as a function of the number of strategic nodes, the number of vessel types and a factor β: `Max_Iterations` = strategic nodes $*$ vessel types $*\beta$. Hence, to find an appropriate maximum number of iterations, an appropriate value of the factor β must be found. This is done by testing 4 different β values on test instances T1, T6, T8 and T10, and comparing the best solutions found with the best solutions from stage 1, where `Max_Iterations` was set to 1000. The β values tested are: 0.20, 0.50, 0.75 and 2.00. Table 8.9 shows the number of iterations for each test instance for each of the respective β-values.

The results from stage 3 testing are presented in Table 8.10 and Table 8.11. For each value of β, the differences in objective function values found for this β-value, relative to the objective function values found in stage 1, are presented. A positive percentage value indicates that the objective function value found for a given β-value is worse (higher) than the objective function value found in stage 1 with 1000 iterations.

Table 8.9: Maximum iterations for each test instance for each β-value.

Test Instance	0.20 Itr [#]	0.50 Itr [#]	0.75 Itr [#]	2.00 Itr [#]
T1	2	6	9	24
T6	8	21	31	84
T8	12	30	45	120
T10	15	39	58	156

Table 8.10: Results from stage 3 calibration testing of the TDG.

Test Instance	0.20 [%]	Time [sec]	0.50 [%]	Time [sec]	0.75 [%]	Time [sec]	2.00 [%]	Time [sec]
T1	4.0	21	0.0	26	0.0	36	0.0	35
T6	0.0	118	0.0	114	0.0	149	0.0	155
T8	0.0	346	0.0	471	0.0	516	0.0	706
T10	0.0	202	0.0	298	0.0	245	0.0	355

Table 8.11: Results from stage 3 calibration testing of the ANG.

Test Instance	0.20 [%]	Time [sec]	0.50 [%]	Time [sec]	0.75 [%]	Time [sec]	2.00 [%]	Time [sec]
T1	3.9	21	0.0	28	0.0	33	0.0	37
T6	0.0	110	0.0	161	0.0	152	0.0	213
T8	0.0	386	0.0	555	0.0	644	0.0	907
T10	0.0	197	0.0	259	0.0	346	0.0	504

The results from the testing show that having a β-factor of 0.2 results in a noticeable decrease in solution quality for the smallest test instance, and this factor is hence too low. For β-values higher than 0.5, the same solution as found in stage 1, is found in all cases for both TDG and ANG. All β-values above 0.5 are hence considered to be sufficient. As can be seen from Table 8.9, these β-values all result in a relatively low number of maximum iterations. This can be seen in connection to the range of α-values for which the GRASP performs the best ([0.00,0.20]), where the GRASP is relatively greedy. The results in Table 8.10 and Table 8.11 also show that the computational time increase with the value of the β-factor. The difference in computational time between $\beta = 0.5$ and $\beta = 0.75$ is relatively small, while the computational time when $\beta = 2.00$ is significantly worse. Compared to $\beta = 0.50$, setting $\beta = 0.75$ results in an increased probability of finding a good solution, without any significant overhead in computational time. Hence, this value has been chosen and is used in all further testing.

8.4 Performance of Solution Methods

Two fundamentally different solution methods are used to solve the DLPOW, a GRASP and a commercial optimization solver. In this section, tests are conducted to evaluate the performance and limitations of these solution methods. In order to conduct such tests, 33 test instances have been generated with various input. All test instances have 120 scenarios and 10 periods, but the size of the test instances vary greatly. The number of strategic nodes vary between 1 - 63, structured in strategic scenario trees of different shapes. Different types of strategic uncertainties are accounted for by the test instances, and the number of stations, vessel types and wind farms vary between test instances.

Some of the generated instances are meant to be fairly easy to solve, while other are meant to test the limitations of the solution methods. A summary of the characteristics of the test instances used is given in Table 8.12.

Table 8.12: Test instances used in the performance testing.

Instance Name	Strategic Uncertainty	#Nodes at Each Stage	#Strategic Nodes	#Vessel Types (#Stations)	#Farms (#Turbines)
D1	Deterministic	(1)	1	4 (2)	2 (200)
D3	Deterministic	(1, 1, 1)	3	4 (2)	2 (200)
D5	Deterministic	(1, 1, 1, 1, 1)	5	4 (2)	2 (200)
D10	Deterministic	(1, 1, 1, ... , 1)	10	4 (2)	2 (200)
D25	Deterministic	(1, 1, 1, ... , 1)	25	4 (2)	2 (200)
F6	Farms	(1, 2, 3)*	6	4 (2)	3 (300)
F10	Farms	(1, 3, 6)*	10	4 (2)	3 (300)
F35	Farms	(1, 3, 6, 10, 15)*	35	4 (2)	3 (300)
F56	Farms	(1, 5, 15, 35)*	56	4 (2)	3 (300)
F56'	Farms	(1, 3, 6, 10, 15, 21)*	56	4 (2)	3 (300)
EP3	Electricity Price	(1, 2)	3	4 (2)	2 (200)
EP4	Electricity Price	(1, 3)	4	4 (2)	2 (200)
EP7	Electricity Price	(1, 2, 4)	7	4 (2)	2 (200)
EP13	Electricity Price	(1, 3, 9)	13	4 (2)	2 (200)
EP21	Electricity Price	(1, 4, 16)	21	4 (2)	2 (200)
EP40	Electricity Price	(1, 3, 9, 27)	40	4 (2)	2 (200)
EP63	Electricity Price	(1, 2, 4, 8, 16, 32)	63	4 (2)	2 (200)
D1.2	Deterministic	(1)	1	6 (3)	2 (200)
D3.2	Deterministic	(1, 1, 1)	3	6 (3)	2 (200)
D5.2	Deterministic	(1, 1, 1, 1, 1)	5	6 (3)	2 (200)
D10.2	Deterministic	(1, 1, 1, ... , 1)	10	6 (3)	2 (200)
D25.2	Deterministic	(1, 1, 1, ... , 1)	25	6 (3)	2 (200)
F6.2	Farms	(1, 2, 3)*	6	6 (3)	3 (300)
F10.2	Farms	(1, 3, 6)*	10	6 (3)	3 (300)
F35.2	Farms	(1, 3, 6, 10, 15)*	35	6 (3)	3 (300)
F56.2	Farms	(1, 5, 15, 35)*	56	6 (3)	3 (300)
F56'.2	Farms	(1, 3, 6, 10, 15, 21)*	56	6 (3)	3 (300)
EP3.2	Electricity Price	(1, 2)	3	6 (3)	2 (200)
EP4.2	Electricity Price	(1, 3)	4	6 (3)	2 (200)
EP7.2	Electricity Price	(1, 2, 4)	7	6 (3)	2 (200)
EP13.2	Electricity Price	(1, 3, 9)	13	6 (3)	2 (200)
EP40.2	Electricity Price	(1, 3, 9, 27)	40	6 (3)	2 (200)
EP63.2	Electricity Price	(1, 2, 4, 8, 16, 32)	63	6 (3)	2 (200)

*The strategic scenario trees for these instances are unbalanced. Their structure is explained in Subection 7.2.2.

8.4.1 Performance of the Optimization Solver

In order to test the performance of the optimization solver, two different methods have been used for solving the 33 test instances. Both methods use the standard settings of the optimization solver and an aggressive cutting strategy. The first method, Exact, solves the original DLPOW problem with a 12 hour time limit (43,200 seconds). The second method, Matheuristic, is a simple matheuristic implemented in the solver. The simple matheuristic consists of two steps, both given a time limit of 12 hours. Step one solves a version of the problem where the integrality constraints of all tactical variables are relaxed, providing a dual bound. In step two, the original problem is solved with the strategic solution fixed to the solution found in step one, providing a primal bound.

Several approaches can be used to find a dual bound in step one, e.g. solving the LP-relaxed problem. The idea behind the method used is to provide the solver with a problem that is easier to solve than the unrelaxed DLPOW, but which may still provide useful information. Since the strategic decision variables are kept integer feasible, and the variables represent feasible real-life actions, it is expected that the approach used in step one of Matheuristic will find a solution of higher quality, and a tighter dual bound, compared to the LP-relaxation.

The results for the two methods are reported in Table 8.13. The best primal and dual bounds found within the time limit, the optimality gap, and the time at which the solver terminates, are reported. None of the tested instances can be solved to optimality (0.01%), and the time reported is hence the time at which the solver reaches its memory limitation. Values marked with a '-' indicate that no solution could be found before the solver reached memory limitations. The reported gap for each test instance is the optimality gap of the primal feasible solution found, calculated as:

$$\text{Optimality Gap [\%]} = \frac{\text{Primal Bound} - \text{Best Dual Bound}}{\text{Primal Bound}} * 100 \tag{8.3}$$

where 'Best Dual Bound' is the globally best dual bound found by either of the two methods. The solutions proven to be within 1.00% of optimality are shown in boldface.

As Table 8.13 shows, only 18 of the 33 tested instances can be handled in its MIP form by the optimization solver, even when the strategic solutions are fixed. The optimization solver is not able to close the optimality gap for any test instance. The smallest optimality gap found is for test instance EP4.2, with a proven optimality gap of 0.11% when solved by the method Exact. Only the 8 smallest and easiest problem instances can be solved to an optimality gap of 1.00% by the method Exact. The method Matheuristic is not able to find a primal feasible solution within 1% of optimality for any test instance. The smallest optimality gap of 5.6% found by this method is for test instance D1. The largest test instance (in terms of size) for which a solution is found within 1% optimality is EP7.2, when using the method Exact. This test instance has 7 strategic nodes, 1,504,440 rows, 5,599,997 columns, and 33,740,110 nonzeros in the constraint matrix. The largest instance for which any feasible solution is found by either method, EP13, has 13 strategic nodes, 2,612,752 rows, 7,602,858 columns and 43,400,476 nonzeros.

The results for Matheuristic in Table 8.13, show that the optimization solver only can handle one more instance when the tactical integrality constraints are relaxed, compared to when solving the original MIP formulation. Furthermore, the results show that the fleet size and mix solution found by Matheuristic is worse than the one found by Exact for 17 of 18 instances. This indicates that the approach used in step one of Matheuristic does not provide good solutions. When inspecting the fleet size and mix solutions found in step one of Matheuristic, it can be seen that the fleets chosen generally are too small compared to what is considered optimal in the original MIP formulation. Furthermore, the results show that Exact finds tighter dual bounds than Matheurinstic for 17 of 18 instances. In conclusion, the method Exact outperforms Matheuristic, as it generally provides both a better solution and better bounds to the problem. However, the results show that, even with an aggressive cutting strategy and a time limitation of 12 hours, finding good solutions with the optimization solver is difficult for anything but small instances.

Table 8.13: Solution values found by the optimization solver.

Instance	Exact				Matheuristic			
Name	Primal [m€]	Dual [m€]	Gap [%]	Time [sec]	Primal [m€]	Dual [m€]	Gap [%]	Time [sec]
D1	**71.66**	71.25	0.6	11,071	75.48	69.19	5.6	14,898
D3	**863.74**	855.27	1.0	10,815	1,008.17	814.22	15.2	14,726
D5	**848.00**	842.02	0.7	10,491	991.89	811.53	15.1	6,167
D10	961.21	831.85	13.5	43,011	1,068.02	822.80	22.1	20,721
D25	-	-	-	-	-	-	-	-
F6	1,088.34	984.58	9.5	13,419	1,043.34	968.79	5.6	6,448
F10	-	956.04	-	990	39,401.78	964.06	97.6	5,913
F35	-	-	-	-	-	-	-	-
F56	-	-	-	-	-	-	-	-
F56'	-	-	-	-	-	-	-	-
EP3	**927.01**	919.49	0.8	3,033	1,138.97	882.36	19.3	19,236
EP4	**900.33**	898.32	0.2	5,475	1,001.64	854.86	10.3	7,125
EP7	886.74	848.06	4.4	17,555	954.25	840.09	11.1	24,234
EP13	968.78	788.81	18.6	952	33,418.48	784.15	97.6	1,385
EP21	-	-	-	-	-	-	-	-
EP40	-	-	-	-	-	-	-	-
EP63	-	-	-	-	-	-	-	-
D1.2	**73.31**	73.19	0.2	8,071	83.01	71.40	11.8	26,281
D3.2	**842.98**	841.73	0.1	12,749	1,107.87	808.20	24.0	15,060
D5.2	881.07	805.46	8.6	33,172	975.90	802.36	17.4	31,194
D10.2	955.68	776.77	18.7	1,494	32,370.55	772.31	97.6	2,244
D25.2	-	-	-	-	-	-	-	-
F6.2	1,138.09	980.95	13.8	30,158	1,187.77	978.89	17.4	24,227
F10.2	-	-	-	-	-	-	-	-
F35.2	-	-	-	-	-	-	-	-
F56.2	-	-	-	-	-	-	-	-
F56'.2	-	-	-	-	-	-	-	-
EP3.2	**880.99**	841.73	4.5	12,749	1,031.10	835.75	18.4	14,486
EP4.2	**874.52**	873.55	0.1	14,314	1,159.82	839.40	24.7	1,887
EP7.2	**860.10**	853.50	0.8	41,803	1,082.63	822.38	21.2	26,756
EP13.2	-	-	-	-	-	-	-	-
EP40.2	-	-	-	-	-	-	-	-
EP63.2	-	-	-	-	-	-	-	-

8.4.2 Performance of the GRASP

To evaluate the performance of the GRASP, 29 of the 33 test instances have been solved with both the TDG and the ANG. These solution methods have been given a time limit of 2 hours (7,200 seconds), and have been run with calibration parameter values as concluded in Section 8.3. Table 8.14 shows the results obtained for the test instances by the two GRASP versions, together with the computational time required to solve each test instance. For the test instances where the time limitation is reached, the GRASP has been interrupted before reaching the maximum number of iterations. The solutions presented are hence the best solutions found by the GRASP within the maximum number of iterations or within the time limit.

As the GRASP offers no guarantee on solution quality, the quality of the GRASP solutions have been assessed by comparing their objective function values to the objective function values found by the solver. The difference (*Diff to Solver*) reported in Table 8.14, show the improvement/deterioration in solution quality when using the GRASP over the optimization solver, and is calculated as:

$$\text{Diff to Solver [\%]} = \frac{\text{Best Objective [GRASP]} - \text{Best Objective [Solver]}}{\text{Best Objective [Solver]}} * 100 \tag{8.4}$$

A negative value indicate that the TDG/ANG finds a lower objective function value, and hence a better solution. The solutions found by the TDG/ANG which are better than the best primal feasible solution found by the solver are shown in boldface. The reported gap in Table 8.14 is the optimality gap for each TDG/ANG solution, which are calculated as shown in Equation 8.3. The optimality gap in the best primal feasible solution found by the solver is reported for comparison in Table 8.14.

Table 8.14: Solution values found by the GRASP.

	Solver	TDG				ANG			
Test Instance	Opt Gap [m€]	Best Objective [m€]	Total Time [sec]	Diff to Solver [%]	Opt Gap [%]	Best Objective [m€]	Total Time [sec]	Diff to Solver [%]	Opt Gap [%]
D1	0.6	71.92	7	0.4	0.9	71.92	6	0.4	0.9
D3	1.0	868.93	42	0.6	1.6	868.93	40	0.6	1.6
D5	0.7	863.10	82	1.8	2.4	862.22	83	-1.7	2.3
D10	13.5	**871.16**	132	-9.4	4.5	**867.45**	171	-9.7	4.1
D25	-	**869.48**	600	-	-	**860.89**	964	-	-
F6	5.6	**1,024.43**	174	-1.8	3.9	**1,022.94**	147	-2.0	3.7
F10	97.6	**1,051.39**	228	-97.3	8.3	**1,051.59**	267	-97.1	8.3
F35	-	**1,116.24**	1,452	-	-	**1,105.87**	5,890	-	-
F56'	-	**1,147.05**	7,200	-	-	**1,131.80**	7,200	-	-
EP3	0.8	938.13	29	1.2	2.0	938.13	32	1.2	2.0
EP4	0.2	914.19	53	1.5	1.7	914.19	59	1.5	1.7
EP7	4.4	**880.91**	79	-0.7	3.7	**880.91**	82	-0.7	3.7
EP13	18.6	**873.57**	130	-9.7	5.1	**873.57**	165	-9.7	5.1
EP21	-	**868.64**	241	-	-	**868.64**	539	-	-
EP40	-	**884.58**	571	-	-	**884.58**	5,671	-	-
EP63	-	**840.93**	7,200	-	-	**839.10**	7,200	-	-
D1.2	0.2	73.84	14	0.7	0.9	73.84	12	0.7	0.9
D3.2	0.1	855.01	75	1.4	1.6	855.01	70	1.4	1.6
D5.2	8.6	**849.34**	162	-3.6	5.2	**847.20**	142	-3.8	4.9
D10.2	18.7	**858.85**	258	-10.1	9.4	**856.40**	337	-10.4	9.2
D25.2	-	**832.19**	1,734	-	-	**826.57**	2,569	-	-
F6.2	13.8	**1,035.16**	350	-9.0	5.2	**1,028.55**	300	-9.6	4.6
F10.2	-	**1,019.97**	549	-	-	**1,015.95**	682	-	-
F56.2	-	**1,113.50**	7,200	-	-	**1,128.50**	7,200	-	-
EP3.2	4.5	893.40	54	1.4	5.8	889.85	52	1.0	5.4
EP4.2	0.1	890.75	87	1.9	1.9	890.75	88	1.9	1.9
EP7.2	0.8	866.61	165	0.8	1.5	866.61	146	0.8	1.5
EP13.2	-	**856.98**	289	-	-	**856.98**	357	-	-
EP40.2	-	**856.06**	1,433	-	-	**856.06**	7,200	-	-

From Table 8.14 it can be seen that both the TDG and ANG are able to find a feasible solution for all the tested instances, within the time limit of 2 hours. Hence, they are both able to solve much larger instances than the optimization solver within a significantly shorter amount of time. However, for the largest instances the time limit is reached in both methods before the procedure has completed the given number of iterations. For the 18 instances where the solver finds a feasible solution, the GRASP finds a better solution for the 8 largest instances, with an improvement in objective function value in the range 1.8% - 97.3%, when compared to the solutions found by the solver. However, the GRASP finds slightly worse solutions for the 10 smaller instances, with a percentage difference in the range 0.4% - 1.9%, compared to the solver.

For the test instances where it is possible to calculate an optimality gap, 16 out of 18 solutions found by TDG/ANG are within 5.8% of optimality. For the remaining 2 instances, both GRASP versions find solutions within 9.4% of optimality, which is significantly lower than what the solver finds. This is considered good, as the non-optimal deployment of the fleet accounts for a part of this gap. In addition, the dual bounds found by the solver might not be very tight, especially for the larger instances. Hence, a high optimality gap in the TDG/ANG solution does not necessarily mean that the solution found is far away from the optimal solution.

All of the test instances (except D1) solve more or less the same problem of finding an optimal fleet to conduct maintenance at 2-3 offshore wind farms, with a planning horizon of 25 years. The problems mainly differ in the amount of vessels to choose from, how often the fleet can be adjusted, and the uncertainty that is accounted for. For this reason, the objective function values for the various test instances should be at approximately the same level (except the instances with uncertainty in stepwise farm development, which have a higher demand for maintenance). When considering the 10 test instances where the solver has not been able to find a solution, it can hence be seen that the objective function values found by the GRASP is at a reasonable level for all test instances. Hence, the results in Table 8.14 indicate that the two GRASP versions find good solutions also for the instances which the solver cannot solve.

When comparing the performance of the TDG to the performance of the ANG, it can be seen that in terms of solution quality, the ANG finds better solutions for 12 of 29 instances. The TDG finds better solutions for 2 of 29 instances. However, the percentage differences in the objective function values found by the two solution methods are relatively small, ranging from 0.00 - 1.34%. In terms of computational time, it can be seen that for smaller instances with up to 13 strategic nodes, the difference in computational time is insignificant. However, for some of the larger instances, the TDG is significantly faster than the ANG. This difference in computational time for large instances is caused by the substantial difference in memory requirements for the two GRASP versions, as the ANG evaluate many more candidate insertions in each iteration. For large instances, the java application hence needs to hold a lot more information in memory while running the ANG than the TDG. As the instances solved by the ANG grow in size, the application starts reaching the memory limitation of the runtime environment, having to spend much time on releasing memory to keep the ANG running. This slows down each iteration of the ANG significantly. This is not a problem for the smaller instances, as these instances have a problem size that allows both GRASP versions to run well within the memory limitation of the runtime environment.

When inspecting the fleet size and mix solutions found by the GRASP, the only solution that is identical to the one found by the optimization solver is the solution for test instance D3. However, when looking at the objective function values (*Best Objective*) reported in Table 8.14, even though both the TDG and the ANG finds the same solution as the solver, the objective function value is worse. This is caused by the fact that the GRASP is not able to deploy the fleet in an optimal way. In order to give a more fair evaluation of the objective function value of the strategic solutions found by the GRASP, two new variations of the solver, S-TDG and S-ANG, have been used. S-TDG/S-ANG uses standard settings in the optimization solver and an aggressive cutting strategy, and solves the problem with the strategic solution fixed to the solution found by the TDG/ANG. Hence, the solution methods find the objective function value of the feet size and mix solution from the TDG/ANG, when deployed optimally.

As the GRASP versions already have proven themselves to provide significantly better objective function values than the solver for the larger instances, only the smallest test instances have been tested. The results from S-TDG and S-ANG are reported in Table 8.15. Objective values found by the S-TDG/S-ANG which are better than the best primal feasible solution found by Exact/Matheuristic are shown in boldface. The improvement in objective function values in the S-TDG/S-ANG, compared to TDG/ANG, are reported as the difference between the two values as a percentage of the TDG/ANG objective. The reported gap is calculated as given in Equation 8.3.

Table 8.15: Objective values of GRASP solutions when fixed in solver.

	S-TDG			S-ANG		
Instance Name	Best Objective [m€]	Diff to TDG [%]	Opt Gap [%]	Best Best [m€]	Diff to ANG [%]	Opt Gap [%]
D1	**71.38**	-0.7	0.2	**71.38**	-0.7	0.2
D3	**857.26**	-1.3	0.2	**857.26**	-1.3	0.2
D5	855.27	-0.9	1.5	857.24	-0.6	1.8
D10	**866.06**	-0.6	4.0	**862.10**	-0.6	2.1
F6	**1,013.13**	-1.1	2.8	**1,016.39**	-0.6	3.1
EP3	**925.77**	-1.3	0.7	**925.77**	-1.3	0.7
EP4	901.55	-1.4	0.4	901.55	-1.4	0.4
EP7	**875.75**	-0.6	3.2	**875.75**	-0.6	3.2
D1.2	73.32	-0.7	0.2	73.32	-0.7	0.2
D3.2	**842.66**	-1.4	0.1	**842.66**	-1.4	0.1
D5.2	**839.73**	-1.1	4.1	**841.50**	-0.7	4.3
D10.2	**852.63**	-0.7	8.9	**849.57**	-0.8	8.6
F6.2	**1,024.35**	-1.0	4.2	**1,018.03**	-1.0	3.6
EP3.2	881.11	-1.4	4.5	877.55	-1.4	4.1
EP4.2	877.85	-1.4	0.5	877.85	-1.4	0.5
EP7.2	**854.91**	-1.4	0.2	**854.91**	-1.4	0.2

The results in Table 8.15 show that when the fleet is deployed optimally, the objective function value for the fleet size and mix solutions found by the TDG/ANG are between 0.6% and 1.4% lower than when the fleet is deployed by the GRASP. This shows that the strategic solutions found by the TDG and ANG are slightly better than what the objective values in Table 8.14 imply. The difference in objective function value found by the TDG and the S-TDG is purely a result of the non-optimal deployment of the fleet in the TDG. The same holds for the S-ANG and ANG. It can hence be seen that, when the cost of non-optimal deployment is removed, the TDG and ANG find a better overall solution than Exact and Matheuristic for 13 of the 18 instances which can be handled by the optimization solver. Furthermore, the results show that, when the tactical deployment decisions are made by the solver, the strategic GRASP solutions are maximum 8.9% away from optimality.

8.4.3 Conclution of Performance Testing

In conclusion, the GRASP is considered as the best solution method for the DLPOW as it outperforms the optimization solver in terms of solution time, and is able to solve significantly larger problems. Furthermore, the GRASP consistently provides good solutions for the DLPOW, and the solutions found by the GRASP is better than those of the solver for most of the tested instances. Even though the GRASP versions do not deploy the fleet optimally, a better solution is found for 19 of the 29 tested instances. Furthermore, when the cost of the non-optimal deployment is removed, a better solution is found for 24 of 29 instances. The two GRASP versions have similar performance, but the ANG finds slightly better solutions for some instances. On the other hand, the total computational time of the ANG is higher than the computational time of the TDG for larger problem instances. The two GRASP variations are hence considered equally good in terms of performance.

8.5 Value of Dual-Level Scenario Tree

The DLPOW has been modelled to capture uncertainties at two decision levels with different time scales. Having strategic uncertainty with embedded tactical uncertainty results in increased complexity and problem size. A question regarding the value of including strategic uncertainty in the model is therefore raised. Such evaluation is conducted through calculations of the Value of Strategic Stochastic Solution (VSSS), as presented in Section 7.5.1. Since test instances are generated with only one type of strategic uncertainty at the time, VSSS calculations need to be conducted for each type of strategic uncertainty separately. The VSSS related to uncertainty in electricity prices and stepwise development of wind farms is evaluated in this section.

As mentioned in Chapter 7, the input data used in this thesis is estimated based on data from the literature. As the availability of good and accurate vessel cost data is limited, the cost parameters used in this thesis may not be appropriate for a real-life wind farm. In order to ensure that the VSSS is not biased towards the choice of input parameter values used in this thesis, different sets of input parameters have been used when calculating VSSS. The motivation behind each input parameter set is explained in the following paragraphs.

A hypothesis is that the total capital cost of acquiring a vessel fleet has consequences for the VSSS. When the capital costs are low, relative to the downtime costs and the variable costs of deploying the fleet, it is suspected that fleet decisions are more guided by the tactical scenarios in a strategic node, rather than future strategic uncertainty. In such cases, the stochastic model might make decisions similarly to a deterministic model, which reduces the VSSS. In order to test how charter costs affect the VSSS, a new input set has been used in the VSSS calculations. In the new input set, the yearly long-term charter costs have been increased, resulting in higher total charter costs. As the cost of short-term chartering vessels in or out should be at an appropriate level relative to the yearly long-term charter cost, the short-term cost and revenue have been increased proportionally to the long-term charter cost. The specific values of the cost parameters used in the new input set is given in Appendix B.3.1.

The size of the lease length discount used to calculate total charter cost of long-term chartering a vessel, as given by Equation (7.5) in Chapter 7, is also expected to influence the VSSS. The discount is given per year, meaning that a yearly discount of 1.5% and a charter contract for 10 years result in a total discount of 15%. If the discount is large, charter contracts with long lease lengths get significantly cheaper yearly charter costs than charter contracts with shorter lease lengths. A high discount may give the effect that it is always cheaper to long-term charter in the root node, where longer contracts are available, rather than chartering at later stages. In such a case, the stochastic model may never choose to wait and adjust the fleet at later stages after uncertainty is revealed. To test how this aspect influence the VSSS, two different values have been used for the lease length discount. To test the effect of increased charter costs and the lease length discount, 4 different sets of input parameters have been used. The difference in the 4 sets of input parameters used are shown in Table 8.16.

Table 8.16: Input sets used in VSSS calculations.

Input Set	Chartering Costs	Lease Length Discount
1	Cost parameters given in Chapter 7	1.5% per year
2	Increased charters costs and revenue.	1.5% per year
3	Same cost parameters as in Input 1.	0.5% per year
4	Same cost parameters as in Input 2.	0.5% per year

To evaluate VSSS, 16 test instances have been used. These 16 test instances can be divided into 4 different types, where all instances of the same type have the same strategic scenario tree, vessel types, wind farms, harbours and offshore stations. The characteristics of the instance types are given in Table 8.17. In the instances with uncertainty in electricity price (electricity instances), the electricity price in a strategic node may deviate with 8% from its direct parent node, giving a maximum possible change in electricity price of 15.36% throughout the planning horizon. In the instances with uncertainty in stepwise development of wind farms (farm instances), a wind farm with 100 turbines is expected to be fully realized in year 10 (stage 2). Figures of the scenario trees for the test instance types are shown in Appendix B.3.2.

Table 8.17: Test instance types used in VSSS Calculations.

Instance Name	Strategic Uncertainty	#Nodes at Each Stage	#Strategic Nodes	#Vessel Types (#Stations)	#Farms (#Turbines)
E1	Electricity Price	(1, 2, 4)	7	4 (2)	2 (200)
E2	Electricity Price	(1, 4, 16)	21	4 (2)	2 (200)
F1	Farms	(1, 2, 3)*	6	4 (2)	3 (300)
F2	Farms	(1, 3, 6)*	10	4 (2)	3 (300)

*The strategic scenario trees for these instances are unbalanced. Their structure is explained in Subection 7.2.2.

As the optimization solver is impractical to use for solving anything but small instances, all test instances needed for calculating VSSS have been solved by the GRASP. The ANG version of the GRASP has been used, as this version gives slightly better solutions, and the computational time needed to solve the test instance types in Table 8.17 is approximately the same for both GRASP versions. The calibration parameters have been set as concluded in Section 8.3. The VSSS results are shown in Table 8.18.

Table 8.18: Value of Strategic Stochastic Solution.

Instance	Input 1		Input 2		Input 3		Input 4	
	[m€]	[%]	[m€]	[%]	[m€]	[%]	[m€]	[%]
E1	0.45	0.1	5.53	0.6	0.00	0.0	0.00	0.0
E2	0.00	0.0	5.30	0.5	0.00	0.0	1.92	0.2
F1	4.02	0.4	5.86	0.5	21.42	2.0	7.71	0.6
F2	2.96	0.3	10.40	0.9	2.62	0.2	1.06	0.1

The results in Table 8.18 show a non-negative VSSS for all test instances, which coheres with Property (7.5.1) discussed in Chapter 7. For 12 of the 16 tested instances, the results show a positive VSSS. The reduction in cost from including strategic uncertainty for these 12 instances range from 0.45 - 21.42 m€. For the electricity instances, the cost reductions range from 0.45 - 5.53 m€, while the similar range for farm instances is 1.06 - 21.42 m€. The results hence indicate that the benefit of accounting for strategic uncertainty in stepwise wind farm development is higher than the benefit of accounting for strategic uncertainty in electricity prices.

Comparing the results for Input 1 with the results for Input 2 shows that an increased charter cost leads to a higher VSSS value for both types of uncertainty. Comparing Input 1 to Input 3, and Input 2 to Input 4, shows that reducing the lease length discount for long-term charter has a negative consequence for the VSSS of the electricity instances. For the farm instances, the effect of reducing the lease length discount is more ambiguous, as it results in an increased VSSS for F1, and a reduced VSSS for F2. Comparing the difference in VSSS between EV1 and EV2, and F1 and F2, the results do not give any clear indication that the VSSS increase with an increasing number of strategic scenarios. However, as all the tested instances have relatively few strategic scenarios, this does not necessarily indicate a low added value of introducing strategic uncertainty.

To analyse whether the positive VSSS results are a consequence of different strategic solutions being made and not a result of the slight tactical in-sample instability, the fleet size and mix solutions of the EEV problem and the SP problem have been compared. For the farm instances, differences caused by the strategic uncertainty can be identified. As an example, in the F2 instance generated on Input 2, the SP chooses to long-term charter a larger fleet in the root node than the EEV. In scenarios where no turbines are built, the SP compensates by chartering out some vessels. The EEV, on the other hand, long-term charters extra vessels when the number of turbines built is larger than expected. This is logical, as the high lease length discount in Input 2 makes it cheaper to long-term charter for 25 years, and rather charter some vessels out when not needed, than to long-term charter for 15 years at the second stage. The SP hence makes a decision in the root node which results in a position that is better for handling future uncertainty. Similar differences can be identified for some of the other farm instances as well. In the electricity instances, the differences in the fleet size and mix decisions made by the SP and the EEV are not unambiguously caused by strategic uncertainty. The EEV makes different choices than the SP, but it is more difficult to conclude that this is a direct consequence of the strategic uncertainty in electricity price.

Having new turbines added to a wind farm results in an increased demand for maintenance, and in this case the need for a larger fleet is evident. However, the effect of the electricity price on the fleet size and mix is more indirect. An increased (decreased) electricity price results in increased (decreased) downtime costs. The variation in electricity price hence only affect the importance of conducting corrective maintenance tasks as soon as possible. If the change in electricity price is large enough, this might offset the preferred size and mix of the fleet. However, the combination of input parameters currently used in the model already gives incentive to have a large fleet that makes it possible to conduct maintenance relatively fast. In order for the SP to make significantly different choices than the EEV, the change in electricity price hence most likely needs to be relatively large. The change in electricity price in the test instances used does not manage to do this, and the tested instances hence do not show a clear benefit of including strategic uncertainty in electricity prices.

In conclusion, the results show that there is a small positive value of accounting for strategic uncertainty. Increasing the chartering costs result in increased VSSS for both types of uncertainties. However, it is not evident from the results how changing the lease length discount affect VSSS. The results indicate a benefit of accounting for uncertainty in stepwise wind farm development. The results are, however, more ambiguous when it comes to the benefit of accounting for uncertainty in electricity prices. Furthermore, the results indicate that several input parameters have a positive or negative effect on the VSSS. As the input parameters used in this thesis are estimated based on data from the literature, and hence may not be accurate for the industry, it makes more sense to analyse the effect of different input parameters and the VSSS when having more appropriate input data.

Chapter 9

Concluding Remarks

The aim of this thesis has been to study the strategic fleet size and mix problem for conducting maintenance at offshore wind farms. A dual-level stochastic model for the DLPOW has been developed, which accounts for both long-term strategic uncertainty and short-term tactical uncertainty, combining decisions with two different time scales in one optimization model. The model supports wind farm owners in making strategic decisions regarding the amount, placement, charter length and types of vessels to long- and short-term charter, to meet maintenance demand throughout the lifetime of a wind farm. To evaluate the quality of these strategic decisions, the model also considers the cost of optimal deployment of the fleet acquired, in light of uncertain demand for maintenance and realizations of weather conditions.

The developed model is the first application of a dual-level stochastic modelling approach for a fleet size and mix problem in offshore wind. The model can account for strategic uncertainties that have not been considered in previously developed optimization models for offshore wind, such as uncertainty related to: long-term trends in electricity prices and subsidy levels, stepwise development of wind farms, and technological development in the vessel industry. A new method, called value of strategic stochastic solution, has been developed to evaluate the added value of including strategic uncertainty in a model that accounts for tactical uncertainty. VSSS calculations conducted for the DLPOW indicate a benefit of accounting for strategic uncertainty.

A scenario generator has been developed to create test instances for the mathematical model. The scenario generator creates tactical scenarios with uncertainty in weather realizations and turbine failures, and strategic scenarios with uncertainty in electricity prices or stepwise development of wind farms. Extensive in- and out-of-sample stability testing has been conducted on the scenario generator to ensure stability. The results from the testing indicate stability in both objective function value and solution, for problem instances with 120 tactical scenarios and 10 periods.

In order to test the applicability of a standard optimization solver on the DLPOW, the solver has been applied both as an exact and as a heuristic solution method. Extensive tests show that the solver is impractical to use for anything but small problem instances due to rapid growth in problem size and memory limitations. The optimization solver is hence considered inadequate for solving problems of realistic sizes. A heuristic solution method, based on the metaheuristic GRASP, has therefore been developed to solve the

DLPOW. This metaheuristic is possibly the second application of a GRASP in a stochastic setting [74], and the first GRASP developed for a dual-level stochastic problem [52]. The reactive GRASP heuristic exploits the block-separable structure of the DLPOW to decompose the problem into a master problem and many independent subproblems. The GRASP constructs strategic fleet size and mix solutions for solving the master problem. A simple Greedy Tactical Heuristic, embedded in the GRASP, solves the subproblems of tactical fleet deployment to evaluate the objective function value of a given fleet solution. Two versions of the reactive GRASP have been developed, where the top-down version is more restricted in terms of how solutions are constructed, compared to the any-node version.

Extensive testing has been conducted on the reactive GRASP. Testing of the Greedy Tactical Heuristic shows that the heuristic performs sufficiently good for its intended use, as it evaluates fleet size and mix solutions similarly as the optimization solver. In addition, tests show that the Greedy Tactical Heuristic finds deployment decisions which are close to the optimal solution in terms of cost. Calibration testing on the reactive GRASP shows that the solution method developed is robust with regards to parameter settings. The performance of the reactive GRASP has been evaluated by comparing solution time and quality from the GRASP to the equivalent values obtained from a standard optimization solver. Test results show that the GRASP consistently provides good solutions for the DLPOW. Compared to the solver, the GRASP finds a better fleet size and mix solution for 24 of the 29 tested instances. Furthermore, the GRASP manages to solve significantly larger instances than the optimization solver, within a considerable shorter amount of time. A comparison of the performance of the two GRASP versions indicates that the TDG is better in terms of computational time, while the ANG finds better solutions.

In order to evaluate the practical applicability of the model developed in this thesis, a suggestion for further work is to test the model on a real-life case from the industry, assessing the models ability to contribute to cost reductions for a real-life wind farm. Another suggestion is to conduct further VSSS testing when more accurate input data is available.

Bibliography

[1] Alvarez, J. F., Tsilingiris, P., Engebrethsen, E. S. and Kakalis, N. M. P. (2011) 'Robust fleet sizing and deployment for industrial and independent bulk ocean shipping companies', *INFOR: Information Systems and Operational Research*, 49(2), pp. 93–107.

[2] Alvarez-Valdesa, R., Parreno, F., Tamarit, J.M. (2008) 'Reactive GRASP for the strip-packing problem', *Computers & Operations Research*, 35(4), pp. 1065–1083.

[3] Anderson, S. (2013) *Comparing Offshore and Onshore Wind.* HSA 10-5 The Economics of Oil and Energy.

[4] Andrawus, J., Watson, J. and Kishk, M. (2007) 'Wind turbine maintenance optimisation: principles of quantitative maintenance optimisation', *Wind Engineering*, 31(2), pp. 101–110.

[5] Andres, C., Miralles, C., and Pastor, R. (2008) 'Balancing and scheduling tasks in assembly lines with sequence-dependent setup times', *European Journal of Operational Research* 187(3), pp. 1212–1223.

[6] Antoniadou, I., Dervilis, N., Papatheou, E., Maguire, A. E. and Worden K. (2015) 'Aspects of structural health and condition monitoring of offshore wind turbines', *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 373 (2035).

[7] Argüello, M., Bard, J. and Yu, G. (1997) 'A GRASP for aircraft routing in response to groundings and delays', *Journal of Combinatorial Optimization*, 1(3), pp. 211–228.

[8] Arroyo, J.E.C., Vieira, P.S., and Vianna, D.S., (2008) 'A GRASP algorithm for the multi-criteria minimum spanning tree problem', *Annals of Operations Research*, 159(1), pp. 125–133.

[9] Bakkehaug, R., Eidem, E. S., Fagerholt, K. and Hvattum, L. M. (2014) 'A stochastic programming formulation for strategic fleet renewal in shipping', *Transportation Research Part E: Logistics and Transportation Review*, 72(), pp. 60–76.

[10] Bakhshi, R., Lei, X., Sandborn, P. A. and Kashanipour, A. R. (2015) *Development of a Maintenance Option Model to Optimize Offshore Wind Farm O&M.* Presented at EWEA Offshore 2015. Copenhagen, Denmark.

[11] Bard, J.F., and Feo, T.A. (1989) 'Operations sequencing in discrete parts manufacturing', *Management Science*, 35(2), pp. 249–255.

[12] Bard, J., and Thaleman, F. () *Offshore Infrastructure: Ports and Vessels. A report of the Off-shore Renewable Energy Conversion platforms – Coordination Action.* Orecca and WaveEnergy Centre.

[13] Bellmore, M. , Bennington, G. and Lubore, S. (1968) 'A maximum utility solution to a vehicle constrained tanker scheduling problem', *Naval Research Logistics Quarterly*, 15(3), pp. 403–412.

[14] Besnard, F. , Fischer K. and Bertling L. (2013) 'A model for the optimization of the maintenance support organization for offshore wind farms', *IEEE Trans Sustain Energy*, 4(2), pp. 443–450.

[15] Besnard, F., Patriksson, M. , Strömberg, A.-B., Wojciechowski, A., Fischer, K. and Bertling, L. (2011) 'A stochastic model for opportunistic service maintenance planning of offshore wind farms', *Proceedings of the IEEE Powertech Conference, June 19–23, Trondheim, Norway.*

[16] Beurskens, J. (2011) *Converting Offshore Wind into Electricity.* The Netherlands: Eburon Uitgeverij BV.

[17] Bianchi, L., Dorigo, M., Gambardella, L. M. and Gutjahr, W. J. (2009) 'A survey on metaheuristics for stochastic combinatorial optimization', *Natural Computing: an international journal*, 8(2), pp. 239-287.

[18] Binato, S., and Oliveira, G.C. (2002) 'A reactive GRASP for transmission network expansion planning', in Ribeiro, C.C., Hansen, P. (Eds.) *Essays and Surveys in Metaheuristics.* Vol. 15. New York: Springer Science & Business Media, pp. 81–100.

[19] Binato, S., and Hery, W.J., Loewenstern, D., Resende, M.G.C. (2002) 'A GRASP for job shop scheduling', in Ribeiro, C.C., Hansen, P. (Eds.) *Essays and Surveys in Metaheuristics.* Vol. 15., pp. 59–79. New York: Springer Science & Business Media.

[20] Birge, J. and Louveaux, F. (2011) *Introduction to stochastic programming.* New York: Springer.

[21] Blum, C., and Roli, A. (2003) 'Metaheuristics in combinatorial optimization: Overview and conceptual comparison', *ACM Computing Surveys*, 35(3), pp. 268-308.

[22] Boudia, M., Louly, M.A.O., and Prins, C. (2007) 'A reactive GRASP and path relinking for a combined production-distribution problem', *Computers & Operations Research* 34(11), pp. 3402–3419.

[23] Bolstad, K. H., and Joshi, M. (2015) *Vessel Fleet Size and Mix for Maintenance at Offshore Wind Farms. A Dual-Level Stochastic Approach.* Project Report. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[24] van Bussel, G.J.W. (2002) 'Offshore Wind Energy, the Reliability Dilemma' in *Proceedings of the International conference.* Berlin, pp. 2-6.

[25] van Bussel, G.J.W. and Zaaijer, M.B. (2001) 'Reliability, Availability and Maintenance Aspects of Large-Scale Offshore Wind Farms, A Concepts Study', in *MAREC 2011: Proceedings of the 2-day International Conference on Marine Renewable Energies, Newcastle, UK, 27-28 March 2001.* Institute of marine engineers.

[26] Bussiéres, F. and Cavaco, P. (2011) *The Wind Energy Operations Maintenance Report.* Wind Energy Update.

[27] Campbell, S. (2014) *Wind Power Monthly: Vessels & Access: A to Z - A maritime guide to offshore wind*. Retrievable from: `http://www.windpowermonthly.com/article/1291038/vessels-access-z-maritime-guide-offshore-wind`. (Accessed: 11.04.2016).

[28] Charles Hodge Photography & Video production (2011) *Greater Gabbard offshore wind farm*, Courtesy of SEE. Retrivable from: `https://flic.kr/p/fpEuWH`. (Accessed: 08.06.2016)

[29] Christiansen, M., Fagerholt, K., Nygreen, B. and Ronen, D. (2007) 'Maritime Transportation', in Barnhart, C. and Laporte, G. (Eds.) *Handbook in Operations Research and Management Science*. Vol. 14. Amsterdam: Elsevier, pp. 189-284.

[30] Christiansen, M., Fagerholt, K. and Ronen, D. (2004) 'Ship routing and scheduling: status and perspectives', *Transportation Science*. 38(1), pp. 1–18.

[31] Consumer.powerswitch () *Spot Prices*. Retrievable from: `https://www.powerswitch.org.nz/powerswitch/site-info/powerswitch-faqs/spot-price` (Accessed: 11.04.2016).

[32] Corbetta, G., Ho, A. and Pineda, I. (2015) *Wind Energy Scenarios for 2030*. The European Wind Energy Association (EWEA).

[33] Corbetta, G., Mbistrova, A., Ho A., Guillet, J. and Pineda I. (2015) *The European offshore wind industry - key trends and statistics 2014*. The European Wind Energy Association (EWEA).

[34] Corbetta, G., Pineda, I. and Wilkes, J. (2015) *Wind in Power: 2014 European Statistics*. The European Wind Energy Association (EWEA).

[35] Corbetta, G., Mbistrova, A., Ho., A., Pineda I., and Ruby K. (2016) *Wind in power. 2015 European statistics*. The European Wind Energy Association (EWEA).

[36] Corbetta, G., Mbistrova, A., Ho., A., Pineda I., and Ruby K. (2016) *The European offshore wind industry - key trends and statistics 2015*. The European Wind Energy Association (EWEA).

[37] Costa, D., Silver and E.A. (1998) 'Tabu Search when noise is present: an illustration in the context of cause and effect analysis', *Journal of Heuristics*, 4(1), pp. 5–23.

[38] Dai, L., Stålhane, M. and Utne, I. B. (2014) 'Routing and Scheduling of Maintenance Fleet for Offshore Wind Farms', *Wind Engineering*, 39(1), pp. 15-30.

[39] Dalgic, Y., Lazakis, I. and Turan, O. (2013) *Vessel charter rate estimation for offshore wind O&M activities*. University of Strathclyde, Department of Naval Architecture & Marine Engineering, UK.

[40] Dantzig, G. (1955) 'Linear programming under uncertainty', *Management Science* 1(3-4), pp. 197–206.

[41] Dantzig G. and Fulkerson, D.R. (1954) 'Minimizing the number of tankers to meet a fixed schedule', *Naval Research Logistics Quarterly*, 1(3), pp. 217–222.

[42] Delmaire, H., Díaz, J.A., Fernandez, E., and Ortega, M. (1999) 'Reactive GRASP and Tabu Search based heuristics for the single source capacitated plant location problem', *INFOR* 37, pp. 194–225.

[43] Dinwoodie, I., Endrerud, O.E., Hofmann, M., Martin, R. and Sperstad, I.B. (2015) 'Reference Case for Verification of Operation and Maintenance Sinmulation Models for Offshore Wind Farms', *Wind Engineering*, 39(1), pp. 1-14.

[44] DNV (2012) *Shipping 2020*. Oslo, Norway: DNV.

[45] Everett, J.L., Hax, A.C., Lewinson, V.A. and Nudds, D. (1972) 'Optimization of a fleet of large tankers and bulkers: a linear programming approach', *Marine Technology*, 9(4), pp. 430–438.

[46] E.ON Climate & Renewables (2012) *E.ON Wind Energy Factbook*. Essen, Germany: E.ON Climate & Renewables.

[47] Esteban, M.D, Diez, J. J., López, J. S., and Negro, V. (2011) 'Why offshore wind energy?' *Renewable Energy*, 36(2), pp. 444-450.

[48] Fagerholt, K. and Lindstad, H. (2000) 'Optimal policies for maintaining a supply service in the Norwegian sea', *Omega*, 28(3), pp. 269–275.

[49] Fagerholt, K., Christiansen, M., Hvattum, L. M., Johnsen, T. A. V. and Vabø, T. J. (2010) 'A decision support methodology for strategic planning in maritime transportation', *Omega*, 38(6), pp. 465–474.

[50] Feo, T. A. and Resende, M. G. C. (1989) 'A probabilistic heuristic for a computationally difficult set covering problem', *Operations Research Letters*, 8(2), pp. 67-71.

[51] Feo, T. A. and Resende, M. G. C. (1995) 'Greedy Randomized Adaptive Search Procedures', *Journal of Global Optimization*, 6(2), pp. 109-133.

[52] Festa, P. and Resende, M. G. C. (2004) 'An annotated bibliography of GRASP', *Operations Research Letters*, 8, pp. 67-71.

[53] Dempster, M. A. H., Fisher, M. L., Jansen, L., Lageweg, B. J., Lenstra, J. K. and Rinnooy Kan, A. H. G. (1981) 'Analytical evaluation of hierarchical planning systems', *Operations Research*, 29(4), pp. 707-716.

[54] Escudero, L.F., Garín, A., Merino, M., and Pérez, G. (2007) 'The value of the stochastic solution in multistage problems', *Top*, 15(1), pp. 48-64.

[55] Forewind (2015) *Forewind Fact Sheet*. Retrievable from: `http://www.forewind.co.uk/uploads/General%20factsheet_February%202015.pdf` (Accessed: 11.04.2016).

[56] Frauenhofer IWES () *Wind Monitor: Distance from shore and water depth*. Retrievable from: `http://windmonitor.iwes.fraunhofer.de/windmonitor_en/4_Offshore/2_technik/4_Kuestenentfernung_und_Wassertiefe` (Accessed: 11.04.2016).

[57] Garrad Hassan & Partners, Tractebel Energy Engineering, Risø National Labratory, Kvaerner Oil & Gas, and Energi & Miljoe Undersoegelser (2001) *Offshore Wind Energy: Ready to Power a Sustainable Europe. Final Report. Concerted Action on Offshore Wind Energy in Europe.* Netherlands: Delft University Wind Energy Research Institute (Duwind).

[58] Gendreau, M., Laporte, G. and Séguin, R. (1996) 'A Tabu Search heuristic for the vehicle routing problem with stochastic demands and customers', *Operations Research*, 44(3), pp. 469–477.

[59] Gill, E. (2013) *Wind Power Offshore: Floating hotels advance offshore construction.* Retrievable from:
`http://www.windpoweroffshore.com/article/1189367/floating-hotels-advance-offshore-construction` (Accessed: 11.04.2016).

[60] Global Wind Energy Council (GWEC) (2016) *Global Wind Statistics 2015.* Brussels: GWEC.

[61] Global Wind Energy Council (GWEC) (2016) *Offshore wind power.*
`http://www.gwec.net/global-figures/global-offshore/` (Accessed: 11.04.2016).

[62] Glover, F. (2000) 'Multi-start and strategic oscillation methods - Principles to exploit adaptive memory', in Laguna, M. and Velarde, J. L. G. (eds.) *Computing Tools for Modeling, Optimization and Simulation.* New York: Springer, pp. 1-23.

[63] Gundegjerde, C. and Halvorsen, I. (2011) *Vessel Fleet Size and Mix for Maintenance Operations of Offshore Wind Farms. A Deterministic Approach.* Specialization Project. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[64] Gundegjerde, C. and Halvorsen, I. (2012) *Vessel fleet size and mix for maintenance of offshore wind farms. A stochastic approach.* Master Thesis. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[65] Gutjahr, W. J. (2004) 'S-ACO: An ant-based approach to combinatorial optimization under uncertainty', in Dorigo, M., Birattari, M., Blum, C., Gambardella, L. M., Mondad, F. and Stützle, T. (eds.) *Ant colony optimization and swarm intelligence.* Berlin: Springer Berlin Heidelberg, pp. 238-249.

[66] Halvorsen-Weare, E. E. and Fagerholt, K. (2011) 'Robust supply vessel planning. In: Network Optimization', in Pahl, J., Reiners, T. and Vob, S. (eds.) *Lecture Notes in Computer Science.* Vol. 6701. Berlin: Springer Berlin Heidelberg, pp. 559–573.

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

[68] Hellemo, L., Midthun, K., Tomasgard, A. and Werner, A. (2013) 'Multi-Stage Stochastic Programming for Natural Gas Infrastructure Design with a Production Perspective', in Gassmann, H. and Ziemba, W. T. (eds.) *Stochastic Programming: Applications in Finance, Energy, Planning and Logistics.* World Scientific Publishing Company, pp. 259-288.

[69] Higle, J. (2005) 'Stochastic programming: Optimization when uncertainty matters', in Smith, J. C. and Greenberg H. J.(Eds.) *Tutorials in Operations Research, Emerging Theory, Methods and Applications.* USA: INFORMS, pp. 30–53.

[70] Hill, R. R., Stinebaugh, J. A., Briand, D., Benjamin, A.S. and Linsday, J. (2008) *Wind Turbine Reliability: A Database and Analysis Approach.* USA: Sandia National Laboratories.

[71] Ho, A. and Mbistrova A. (2015) *The European offshore wind industry - key trends and statistics 1st half 2015.* The European Wind Energy Association (EWEA).

[72] Hoff, A., Andersson, H., Christiansen, M., Hasle, G. and Løkketangen, A. (2010) 'Industrial aspects and literature survey: fleet composition and routing', *Computers & Operations Research* 37(12), pp. 2041–2061.

[73] Hvattum, L. M. (2016) 'An introduction to metaheuristics', Lecture note from lecture in *TIØ4150 — Industrial Optimization and Decision Support*. Presented at Norwegian University of Science and Technology, Trondheim, Norway.

[74] Hvattum, L. M., Løkketangen, A. and Laporte, G. (2009) 'Scenario Tree-Based Heuristics for Stochastic Inventory-Routing Problems', *INFORMS Journal on Computing*, 21(2), pp. 268-285.

[75] Hvide Sande () *Wind Transporter*. Retrievable from:
http://hvsa.dk/portfolio_page/wind-transporter (Accessed: 11.04.2016).

[76] International Energy Agency (IEA) (2015) *World Energy Outlook Special Report. Energy and Climate Change*. OECD/IEA.

[77] Jin, T., Ding, Y., Guo, H. and Nalajala, N. (2012) *Managing wind turbine reliability and maintenance via performance-based contract*. Presented at IEEE Power and Energy Society General Meeting, July 22–26. San Diego, California, USA.

[78] De Jonghe, C., Hobbs, B. and Belmanns, R. (2011) *Integrating short-term demand response into long-term investment planning*. Faculty of Economics, University of Cambridge.

[79] Kaiser, M. J. and Snyder, B. (2005) *Offshore Wind Energy Installation and Decommissioning Cost Estimation in the U.S. Outer Continental Shelf*. Lousiana: Energy Research Group LLC.

[80] Kaut M., Midthun K. T. , Werner A. S. , Tomasgard A. , Hellemo L. and Fodstad, M. (2012) *Multi-horizon stochastic programming*. Optimization Online. Retrievable from:
http://www.optimization-online.org/DB$_{H}TML$/2012/08/3551.*html*

[81] Kaut, M. and Wallace, S. W. (2007) 'Evaluation of scenario-generation methods for stochastic programming', *Pacific Journal of Optimization*, 3(2), pp. 257-271.

[82] Krohn, S., Morthorst, P.-E. and Awerbuch, S. (2009) *The Economics of Wind Energy*. Denmark: The European Wind Energy Association (EWEA).

[83] Kumar, S. (2014) *Marine Insight: 7 Technologies That Can Change The Future of Shipbuilding*. Retrievable from:
http://www.marineinsight.com/marine/marine-news/headline/shipbuilding-technologies/ (Accessed: 11.04.2016).

[84] Lai, M.F. and Lo, H.K. (2004) 'Ferry service network design: optimal fleet size, routing, and scheduling', *Transportation Research Part A: Policy and Practice*, 38(4), pp. 305-328.

[85] Lawler, E. L., and Wood, D. E. (1966) 'Branch-and-bound methods: A survey' *Operations Research*, 14(4), pp. 699-719.

[86] Lindqvist, M., Lundin, J. (2010) *Spare part logistics and optimization for wind turbines: methods for cost-effective supply and storage*. Master thesis. Department of Information Technology, Division of Systems and Control, Uppsala University, Sweden.

[87] London Array () *The Project*. Retrievable from:
http://www.londonarray.com/the-project-3/ (Accessed: 11.04.2016).

[88] Louveaux, F. V. (1986) 'Multistage stochastic programs with block-separable recourse', in Prekopa, A. and Wets, R. J. B. (eds.) *Stochastic Programming 84 Part II, Mathematical Programming Studies 28*. Berlin: Springer Berlin Heidelberg, pp. 48-62.

[89] Lourenço, H.R., Serra, D (2002) 'Adaptive approach heuristics for the generalized assignment problem', *Mathware Soft Computing*, 7, pp. 1-15.

[90] Lynn, P. A. (2011) *Onshore and Offshore Wind Energy: An Introduction.* John Wiley & Sons.

[91] Maples, B., Saur, G., Hand, M., van de Pietermen, R. P. and Obdam, T. (2013) *Installation, Operation, and Maintenance Strategies to Reduce the Cost of Offshore Wind Energy.* USA: National Renewable Energy Laboratory (NREL).

[92] Mast, E., Echavarria, E., Sevilla, F. and Redfern, R. (2015) *Helideck and accommodation facilities on offshore platforms for wind farms.* Arnhem, The Netherlands: DNV GL Energy.

[93] Meng, Q. and Wang, T. (2010) 'A chance constrained programming model for short-term liner ship fleet planning problems', *Maritime Policy & Management*, 37, pp. 329–346.

[94] Meng, Q. and Wang, T. (2011) 'A scenario-based dynamic programming model for multi-period liner ship fleet planning', *Transportation Research Part E*, 47, pp. 401–413.

[95] Meng, Q., Wang, T. and Wang, S. (2012) 'Short-term liner ship fleet planning with container transshipment and uncertain container shipment demand', *European Journal of Operational Research*, 223, pp. 96–105.

[96] Milborrow, D. (2010) *Wind Power Monthly: Breaking down the cost of wind turbine maintenance.* Retrievable from:
`http://www.windpowermonthly.com/article/1010136/breaking-down-cost-wind-turbine-maintenance` (Accessed: 24.02.2016).

[97] Murotsu, Y. and Taguchi, K. (1975) 'Opimization of ship fleet-size', *Bulletin of University of Osaka Prefecture, Series A: Engineering and Natural Sciences* 23, pp. 171–192

[98] Myklebust, J. (2010) *Techno-economic modelling of value chains based on natural gas – with consideration of CO2 emissions.* PhD thesis. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[99] Oddfjell Wind AS () *Special Service Vessels for Wind, Oil & Gas Industry.* Retrievable from:
`http://www.odfjellwind.com/index.php/fob-swath-photos` (Accessed: 11.04.2016).

[100] O'Donnel, D., Srbinovsky, B., Murphy, J., Popovici, E. and Pakrashi, V. (2015) 'Sensor Measurement Strategies for Monitoring Offshore Wind and Wave Energy Devices', *Journal of Physics: Conference Series*, 628.

[101] OffshoreWIND.biz () *Tia Elizabeth - Mercurio Plastics Shipyard.* Retrievable from:
`http://vessels.offshorewind.biz/vessels/tia-elizabeth` (Accessed: 11.04.2016).

[102] OffshoreWIND.biz () *Bond Air Services Starts Operations for Greater Gabbard Offshore Wind Farm, UK.* Retrievable from:
`http://www.offshorewind.biz/2012/10/22/bond-air-services-starts-operations-for-greater-gabbard-offshore-wind-farm-uk/` (Accessed: 11.04.2016).

[103] OffshoreWIND.biz () *Atlantic's Mothership to Undergo DP2 Fitting.* Retrievable from:
`http://www.offshorewind.biz/2015/04/08/atlantics-mothership-to-undergo-dp2-fitting/` (Accessed: 11.04.2016).

[104] OffshoreWIND.biz () *Offshore Wind Installation Vessels Market to Hit USD 2.93 Bln by 2020*. Retrievable from: `http://www.offshorewind.biz/2014/12/10/offshore-wind-installation-vessels-market-to-hit-usd-2-93-bln-by-2020/` (Accessed: 11.04.2016).

[105] Pantuso, G., Fagerholt, K. and Hvattum L.M (2014) 'A survey on maritime fleet size and mix problems', *European Journal of Operational Research*, 235(2), pp. 341-349.

[106] Pantuso, G., Fagerholt, K. and Wallace, S. W. (2015) 'Solving Hierarchical Stochastic Programs: Application to the Maritime Fleet Renewal Problem', *INFORMS Journal on Computing*, 27(1), pp. 89-102.

[107] Pantuso, G., Fagerholt, K. and Wallace, S. W. (2015)'Uncertainty in Fleet Renewal: A Case from Maritime Transportation', *Transportation Science, Articles in Advande*, pp. 1-8.

[108] Pesenti, R. (1995) 'Hierarchical resource planning for shipping companies', *European Journal of Operational Research*, 86, pp. 91–102.

[109] Phillips, J., Fitch-Roy, O., Reynolds, P. and Gardner, P. from GL Garrad Hassan (2013) *A Guide to UK Offshore Wind Operations and Maintenance*. Scottish Enterprise and The Crown Estate.

[110] Power Generation Services, Siemens (2015) *Siemens: Siemens and Esvagt christen wind industry's first offshore Service Operation Vessels*. Retrievable from: `http://www.siemens.com/press/PR2015060263PSEN` (Accessed: 11.04.2016).

[111] Prais, M. and Rebeiro, C. C. (2000) 'Reactive GRASP: An Application to a Matrix Decomposition Problem in TDMA Traffic Assignment', *INFORMS Journal on Computing*, 12(3), pp. 164-176.

[112] PricewaterhouseCoopers Advisory (PWC) & Ecofys Netherlands B.V. (2015) *Subsidy Schemes and Tax Regimes*. Netherlands: TKI Wind op Zee.

[113] van de Pieterman, R. P., Braam, H., Obdam, T. S., Rademakers, L. W. M. and van der Zee, T. J. J. (2011) *Optimisation of maintenance strategies for offshore wind farms*. Presented at The Offshore 2011 conference. Amsterdam, The Netherlands.

[114] Rademakers, L. W. M., Braam, H., Zaaijer, M. B. and van Bussel, G. J. W. (2003) *Assessment and optimisation of operation and maintenance of offshore wind turbines*. Presented at the European Wind Energy Conference (EWEC) 2003. Madrid, Spain.

[115] Rademakers, L. W. M., Obdam, T., Braam, H. and Eecen, P. (2007) *Estimating costs of operation & maintenance for offshore wind farms*. Presented at the European Offshore Wind Energy Conference 2007. Berlin, Germany.

[116] Raknes,N. T. and Ødeskaug, K. (2015) *Optimal Scheduling of Maintenance Tasks and Routing of a Joint Vessel Fleet for Multiple Offshore Wind Farms*. Master Thesis. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[117] Resende, M. G. C. and Ribeiro, C. C. (2010) 'Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations, and Applications', in Gendreau, M. and Potvin, J.-Y. (eds.) *Handbook of Metaheuristics, International Series in Operations Research & Management Science 146*. New York: Springer, pp. 283-319.

[118] Ribrant, J. (2006) *Reliability performance and maintenance - A survey of failures in wind power systems*. Master Thesis. KTH School of Electrical Engineering, Stockholm.

[119] Ribrant J. and Bertling, L. (2007) 'Survey of failures in wind power systems with focus on Swedish wind power plants during 1997-2005', in *Power Engineering Society General Meeting*. IEEE, pp. 1-8.

[120] Ronen, D. (1983) 'Cargo ships routing and scheduling: survey of models and problems', *European Journal of Operational Research*, 12(2), pp. 119–126.

[121] Schütz, P., Tomasgard, A. and Ahmed, S. (2009) 'Supply chain design under uncertainty using sample average approximation and dual decomposition', *European Journal of Operational Research*, 199(2), pp. 409–419.

[122] Schwartz, N.L. (1968) 'Discrete programs for moving known cargos from origins to destinations on time at minimum bargeline fleet cost', *Transportation Science*, 2, pp. 134–145.

[123] Shafiee, M. (2015) 'Maintenance logistics organization for offshore wind energy: Current progress and future perspectives', *Renewable Energy*, 77(), pp. 182–193.

[124] Sheng, S. (2013) *Report on Wind Turbine Subsystem Reliability – A Survey of Various Databases*. USA: National Renewable Energy Laboratory.

[125] Shyshou, A., Gribkovskaia, I. and Barceln, J. (2010) 'A simulation study of the fleet sizing problem arising in offshore anchor handling operations', *European Journal of Operational Research*, 203(1), pp. 230-240.

[126] Siemens AG (2015) *The offshore workhorse – redefined*. Germany: Siemens AG (Order No. WPOF-B10006-00-7600).

[127] Sigurd, M. M., Ulstein, N. L., Nygreen, B. and Ryan, D. M. (2005) 'Column Generation', in *Ship Scheduling With Recurring Visits and Visit Separation Requirements*. New York: Springer, pp. 225-245.

[128] Skaar, V. (2014) *Optimization of routing and scheduling for performing maintenance at offshore wind farms*. Master Thesis. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[129] Sönmez, E., Kekre, S., Scheller-Wolf, A. and Secomandi, N. (2013) 'Strategic analysis of technology and capacity investments in the liquefied natural gas industry', *European Journal of Operational Reseach*, 226(1), pp. 100–114.

[130] Sosnowska, D. (2000) 'Optimization of a simplified fleet assignment problem with metaheuristics: Simulated annealing and GRASP', in Pardalos, P. M. (eds.) *Approximation and Complexity in Numerical Optimization*. New York: Springer, pp. 477-488.

[131] Späth, N. (2015) *DNV GL: Update on "Shipping 2020": DNV GL sees hybrid propulsion and connectivity as emerging trends*. Retrievable from:
`https://www.dnvgl.com/news/update-on-shipping-2020-dnv-gl-sees-hybrid-propulsion-and-connectivity-as-emerging-trends-25931` (Accessed: 11.04.2016).

[132] Tian, Z., Jin, T., Wu, B. and Ding, F. (2011) Condition based maintenance optimization for wind power generation systems under continuous monitoring', *Renewable Energy*, 36(5), pp. 1502-1509.

[133] Umoe Mandal () *Wave Craft.* Retrievable from:
`http://www.um.no/web/um200.nsf/pages/WaveCraft` (Accessed: 11.04.2016).

[134] Vefsnmo, H. M. (2012) *Determining the Optimal Vessel Fleet for Maintenance of Offshore Wind Farms.* Master Thesis. Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management, Trondheim, Norway.

[135] Verderame, P., Elia, J., Li, J. and Floudas, C. (2010) 'Planning and scheduling under uncertainty: a review across multiple sectors', *Industrial & engineering chemistry research*, 49(9), pp. 3993–4017.

[136] World Maritime News () *Dutch Post Workboats Develops New Vessel for Offshore Wind Farm Industry.* Retrievable from:
`http://worldmaritimenews.com/archives/62389/dutch-post-workboats-develops-new-vessel-for-offshore-wind-farm-industry/` (Accessed: 27.05.2016).

[137] Zeng, Q. and Yang, Z. (2007) 'Model integrating fleet design and ship routing problems for coal shipping', in Shy, Y. et al. (Eds.) *Computational Science–ICCS 2007, 7th International Conference, Beijing China, May 27-30, 2007, Proceedings, Part III.* Berlin: Springer Berlin Heidelberg, pp. 1000-1003.

[138] 4C Offshore (2015).
textitBard Offshore 1 Offshore Wind Farm. Retrievable from:
`http://www.4coffshore.com/windfarms/bard-offshore-1-germany-de23.html`
(Accessed: 11.04.2016).

[139] 4C Offshore (2013) *4C Offshore: Wind Farm Service Vessels - An Overview.* Retrievable from:
`http://www.4coffshore.com/windfarms/wind-farm-service-vessels---an-overview-aid2.html` (Accessed: 11.04.2016).

Appendix A

Mathematical Formulation

This appendix contains a plain version of the mathematical formulation of the DLPOW, without explanations. It is added for readers who prefer to read the whole model without interruptions. A thorough explanation of the mathematical formulation is presented in Section 5.3.

Objective function

$$\min \quad z = \sum_{n\in N}\sum_{v\in V}\frac{B_n^S}{Z_n}\left(\sum_{l\in L_{nv}} C_{nvl}^{TC} x_{nvl} + C_{nv}^F w_{nv} + \sum_{q\in Q}(C_{nqv}^{ST} y_{nqv}^{IN} - R_{nqv} y_{nqv}^{OUT})\right) \tag{A.1}$$

$$+\sum_{d\in D} C_d^D \delta_d \tag{A.2}$$

$$+\sum_{n\in N}\sum_{q\in Q}\sum_{s\in S_{nq}}\frac{B_{nqs}^T}{Z_n}\left(\sum_{p\in P_{nqs}}\sum_{f\in F_n}\left[\sum_{m\in M_{nqsf}}\sum_{v\in V_m} C_v^V t_{nqspfmv} + \sum_{v\in V} C_v^V u_{nqspfv} T_{fv}^T\right.\right. \tag{A.3}$$

$$\left. + \sum_{m\in M_{nqf}^{PREV}}\sum_{v\in V_m} C_{nqspf}^{DTP} t_{nqspfmv} + \sum_{m\in M_{nqsf}^{CORR}} C_{nqspfm}^{DTC}\gamma_{nqspfm}\right] \tag{A.4}$$

$$\left. + \sum_{f\in F_n}\sum_{m\in M_{nqsf}} C_m^P \beta_{nqsfm}\right) \tag{A.5}$$

Constraints for Strategic Nodes

$$\sum_{l \in L_{nv}} x_{nvl} + w_{a(n)v} - \sum_{n' \in A_n} x_{n'vt(n)} = w_{nv}, \qquad n \in N \backslash \{1\}, v \in V \tag{A.6}$$

$$\sum_{l \in L_{nv}} x_{nvl} = w_{nv}, \qquad n = 1, v \in V \tag{A.7}$$

$$y_{nqv}^{OUT} \leq w_{nv}, \qquad n \in N, q \in Q, v \in V \tag{A.8}$$

$$\sum_{v \in V_d} G_{vd}(w_{nv} + y_{nqv}^{IN}) \leq M_d^D \delta_d, \qquad n \in N, q \in Q, d \in D \tag{A.9}$$

Constraints for Tactical Scenarios

$$\sum_{p \in P_{nqs}} \sum_{v \in V_m} t_{nqspfmv} \geq T_m^M (1 - \beta_{nqsfm}), \qquad n \in N, q \in Q, s \in S_{nq}, f \in F_n, m \in M_{nqsf} \tag{A.10}$$

$$\sum_{m \in M_{nqsf}} t_{nqspfmv} \leq E_v M_v^{CREW} (T_v^{MAX} - T_{fv}^T) u_{nqspfv},$$
$$n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, v \in V_m \tag{A.11}$$

$$\sum_{f \in \{0\} \cup F_n} u_{nqspfv} = (w_{nv} + y_{nqv}^{IN} - y_{nqv}^{OUT}), \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V \tag{A.12}$$

$$(M_{vk}^K - U_{nqspk}) \sum_{m \in M_{nqsf}} \sum_{f \in F_n} t_{nqspfmv} \geq 0, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V_m \tag{A.13}$$

$$\sum_{p' = \{(p+1), \ldots, |P_{nqs}|\}} \sum_{v \in V_m} t_{nqsp'fmv} \leq T_m^M (1 - \gamma_{nqspfm}),$$
$$n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, m \in M_{nqsf}^{CORR} \tag{A.14}$$

$$\sum_{p \in P_{nqs}} (\gamma_{nqspfm} + \beta_{nqsfm}) = 1, \qquad n \in N, q \in Q, s \in S_{nq}, f \in F_n, m \in M_{nqsf}^{CORR} \tag{A.15}$$

Non-negativity, integrality and binary constraints

$$x_{nvl} \geq 0 \quad \text{and integer}, \qquad n \in N, v \in V, l \in L_{nv} \tag{A.16}$$

$$w_{nv} \geq 0 \quad \text{and integer}, \qquad n \in N, v \in V \tag{A.17}$$

$$y_{nqv}^{IN} \geq 0 \quad \text{and integer}, \qquad n \in N, q \in Q, v \in V | L_{nv} \neq \emptyset \tag{A.18}$$

$$y_{nqv}^{OUT} \geq 0 \quad \text{and integer}, \qquad n \in N, q \in Q, v \in V \tag{A.19}$$

$$u_{nqspfv} \geq 0 \quad \text{and integer}, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V, f \in F_n \cup \{0\} \tag{A.20}$$

$$t_{nqspfmv} \geq 0, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, v \in V_m, f \in F_n, m \in M_{nqsf} \tag{A.21}$$

$$\gamma_{nqspfm} \in \{0, 1\}, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, m \in M_{nqsf}^{CORR} \tag{A.22}$$

$$\beta_{nqsfm} \in \{0, 1\}, \qquad n \in N, q \in Q, s \in S_{nq}, p \in P_{nqs}, f \in F_n, m \in M_{nqsf} \tag{A.23}$$

$$\delta_d \in \{0, 1\}, \qquad d \in D \tag{A.24}$$

Appendix B

Additional Info on Test Instances

In this appendix, some additional information about the test instances used in the computational study is given.

B.1 Duration of strategic nodes

Table B.1 provides information about the duration of the strategic nodes in the various test instances used in the computational study. All test instances with the same number of stages in the strategic scenario tree have the same duration of strategic nodes at each stage in the tree. As an example, a scenario tree with 3 stages hence has strategic nodes at year 0, year 10 and year 24.

Table B.1: Duration of strategic nodes in test instances.

#Stages	Start Year for Each Stage
1	(0)
2	(0, 24)
3	(0, 10, 24)
4	(0, 5, 10, 24)
5	(0, 5, 10, 15, 24)
6	(0, 2, 5, 10, 15, 24)
7	(0, 2, 5, 10, 15, 20, 24)
10	(0, 2, 4, 6, 8, 10, 12, 14, 18, 24)
25	(1, 2,..., 23, 24)

B.2 Fleet Solutions used in Tactical Testing

The 10 different fleet size and mix solutions used to evaluate the Greedy Tactical Heuristic in Subsection 8.2.3, is given in Table B.2. The two vessel types used are the ones given in Table 7.1, belonging to station 2 in Table 7.3. All long-term chartered vessels are charted for 1 year at the cost of a 10 year charter contract. No vessels are chartered out.

Table B.2: Fleet solutions used in testing of Greedy Tactical Heuristic.

Fleet Number	Long-term Charter		Short-Term Charter			
			Summer		Winter	
	Vessel 1	Vessel 2	Vessel 1	Vessel 2	Vessel 1	Vessel 2
1	2	2	1	0	0	0
2	2	2	2	0	0	0
3	3	2	1	0	0	0
4	2	1	2	0	0	1
5	2	1	1	0	0	1
6	3	3	1	1	0	0
7	2	1	2	1	0	0
8	2	0	1	0	0	2
9	2	0	2	2	0	0
10	0	0	0	0	0	0

B.3 Extra info on Test Instances for VSSS Tests

B.3.1 Vessel Cost Input

The two sets of vessel cost input used in VSSS testing are given in Table B.3. Input set 1 gives the original input data used in all computational tests throughout this master thesis. Input set 2 gives a case where the vessel acquisition costs have been increased, and is the data set used for Input 2 and Input 4 in the VSSS testing.

Table B.3: Vessel cost parameters used in VSSS testing.

Input Set	Vessel Number	1 year [m€/yr]	Summer (in) [m€]	Summer (out) [m€]	Winter (in) [m€]	Winter (out) [m€]
1	1	1.1	0.72	0.33	0.69	0.23
	2	1.8	1.17	0.54	1.12	0.44
2	1	2.2	1.5	0.44	1.44	0.34
	2	3.6	2.4	0.72	2.34	0.62

B.3.2 Scenario Trees

The value of strategic stochastic solution is calculated for test instances with uncertainty in electricity price or stepwise development of wind farms. The scenario trees used in test instance types E1 and F2 are depicted in Figure B.1 and Figure B.2, respectively. Figure B.1 shows how the electricity price, of €78.0 per MWh in the root node, varies throughout the scenario tree. The percentages given for each scenario at the end of the planning horizon indicate the total change in electricity price relative to the original electricity price in the root node.

Figure B.2 shows a strategic scenario tree with uncertainty in stepwise wind farm development, where 100 turbines are planned to be realised in year 10 (stage 2). The number in each strategic node indicates the number of turbines at the farm in that respective node. The percentage next to each strategic node gives the percentage of the 100 planned turbines which have been built in that strategic node. The percentages given for each strategic scenario at the end of the planning horizon give the total fraction of the planned wind farm that is built.

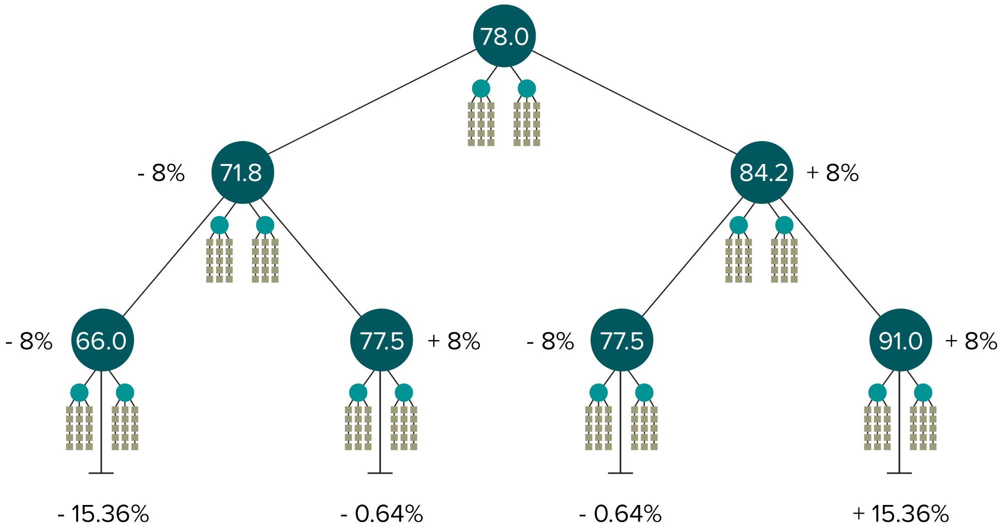


Figure B.1: Scenario tree for test instances of type E1.

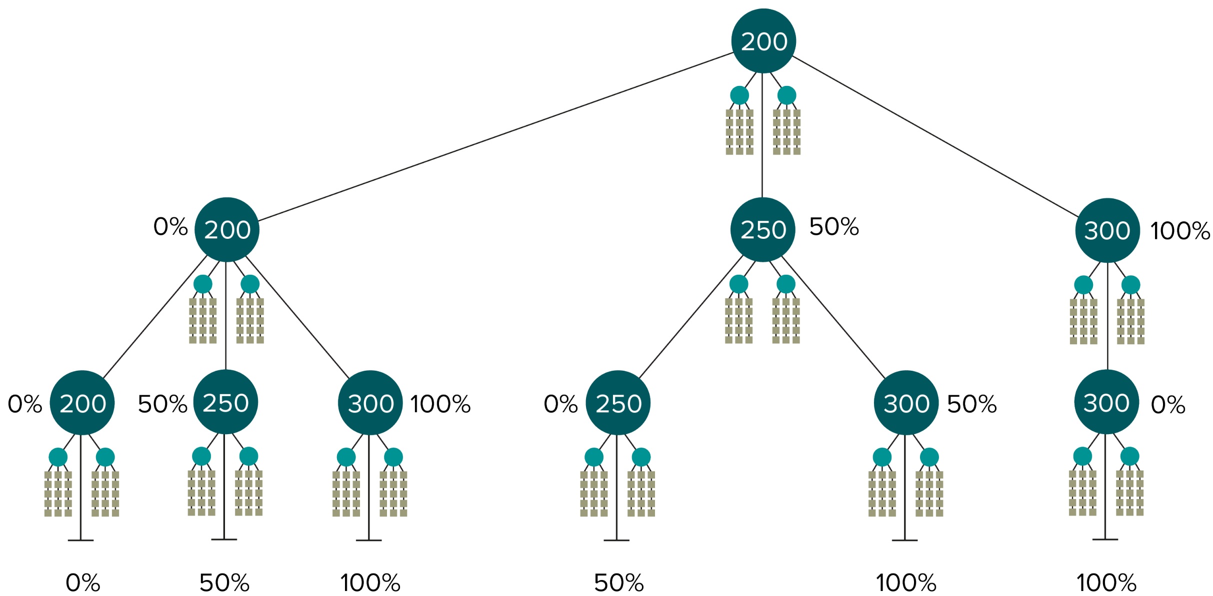


Figure B.2: Scenario tree for test instances of type F2.