A Multi-timescale Framework for Maintenance Planning at Offshore Wind Farms

D. Moros[a,b,*], N. Berrabah[b], I.G. Ashton[c], I. Lazakis[d], K.D. Searle[e]

[a]*Industrial CDT in Offshore Renewable Energy, School of Engineering, The University of Edinburgh, Edinburgh, EH8 9YL, UK*
[b]*Renewables Research & Development, EDF UK, 80 Victoria Street, London, SW15 5JL, UK*
[c]*College of Engineering, Mathematics and Physical Sciences, University of Exeter, Penryn Campus, Cornwall, TR10 9FE, UK*
[d]*Department of Naval Architecture, Ocean and Marine Engineering, University of Strathclyde, Glasgow, G1 1XQ, UK*
[e]*School of Mathematics and Maxwell Institute for Mathematical Sciences, University of Edinburgh, James Clerk Maxwell Building, Kings Buildings, Mayfield Road, Edinburgh, EH9 3FD, UK*

Abstract

To facilitate the continued growth of offshore wind farm developments, operations and maintenance costs, estimated at 30% of the lifetime costs of wind farms, must be reduced. To help enable this reduction a framework is presented for minimising the costs, including lost revenues, of planning maintenance across tactical and operational timescales. The tactical timescale is addressed by solving a long-term job allocation problem, formulated as a mixed-integer linear program, that allocates jobs to time periods within a year. As new maintenance jobs arise and forecasts are updated the long-term job allocation problem is resolved and the maintenance teams choose a new tactical plan. A maintenance routing and scheduling problem is then solved for the jobs assigned to the relevant time period by the tactical model, producing a schedule and set of routes that the maintenance teams can use to plan their operational activities. This model is formulated as a mixed-integer linear program and adaptive large neighbourhood search meta and matheuristic solution methods are proposed for efficient computation. Computational experiments verify that the proposed metaheuristic method is able to obtain solutions on the order of minutes and offer up to 49% improvements in maintenance costs over current practices. Further computational experiments reveal that the framework, applied to a full year of historic maintenance jobs, could lead to an improvement of 66% over the historical baseline. Implementing computational methods for planning maintenance will help to reduce wind farm revenue losses and aid the work of maintenance teams.

Keywords: Maintenance Optimisation; Scheduling; Routing; Matheuristic; Metaheuristic; Adaptive Large Neighbourhood Search; Offshore Wind Farm; Operations and Maintenance

1. Introduction

Offshore wind developments continue to grow globally at a rapid pace, with growth estimates of 630GW of total capacity between 2022 and 2050 [12]. To facilitate this rapid growth, costs must continue to be reduced. *Operations and Maintenance* (O&M) costs, estimated at 30% of the lifetime costs of a wind farm [4], offer opportunities for cost reductions. With *Offshore Wind Farms* (OWFs) increasing in size, moving further offshore and the first floating offshore wind farms deploying, it is becoming increasingly important to develop and implement O&M strategies that can reduce the associated O&M costs. Producing maintenance schedules and routes that can be used by operators to minimise the costs and lost revenue of performing maintenance tasks could facilitate these required cost reductions.

Performing maintenance at OWFs brings unique challenges compared to regular machine maintenance scheduling. Primarily, the weather conditions have a large impact on both the ability to perform maintenance and the cost of revenue losses associated with performing maintenance. Technicians must travel from an O&M base, typically a port, to the OWF and safely transfer from a vessel to a *Wind Turbine* (WT). Transfers and maintenance can only happen during suitable weather

*Corresponding author: d.moros@sms.ed.ac.uk

windows where the conditions are calm enough. Additionally, the revenue generation of a WT is directly linked to the wind speed, so that maintenance should be scheduled during periods of low wind speed to minimise losses from turbine downtime. In addition to weather conditions, maintenance is dependent on the resources available, such as spare parts and qualified technicians. Finally, travel distances at OWFs can be very large, sometimes with distances over 100km from the O&M base to the wind farm, therefore the time and costs related to travel to and from the OWF and between WTs must also be considered.

Maintenance is performed by one of three parties: the OWF operator; the WT manufacturer; or a *Maintenance Service Provider* (MS). Each have different objectives. The objective of the operator is to maximise their profit, by minimising the revenue losses in operating the wind farm, whilst the aim of the manufacturer or MS is to minimise their cost of performing maintenance whilst maintaining a minimum level of acceptable service. This article takes the perspective of the operator. This involves minimising both the costs to perform maintenance and revenue losses from turbine downtime and is a decision-making problem. Decision making occurs across three different timescales, grouped by Shafiee [19] as:

- Strategic – Long-Term (5+ years). Considering overall maintenance strategy; staff requirements and fleet make up.
- Tactical – Medium-Term (months to several years). Considering staff resource planning; vessel leasing/purchasing; spare parts stock planning; maintenance scheduling.
- Operational – Short-Term (days to months). Considering weather conditions; staff availability; maintenance task scheduling and routing of vessels.

Most research has focused on the strategic aspect of maintenance planning [8]. After commissioning, there is little scope for influencing the strategic decisions as these are made at the start of the wind farm life. Therefore, this article focuses on the tactical and operational aspects of planning.There are three decisions that the operator is most concerned with on the tactical and operational timescale, what maintenance needs to be done; when should the maintenance be done; and how should the vessel be routed? This article focuses on the second two questions, in the tactical and operational time scales, which can be summarised as a long-term job allocation problem and a *Maintenance Routing and Scheduling Problem* (MRSP). A framework for combing the tactical and operational decision making in a way that minimises both the cost to perform maintenance and revenue losses from WT downtime at an OWF is presented in this article.

The main contributions of this article are summarised as follows:

1. A novel framework for planning maintenance over tactical and operational timescales that combines yearly planning with daily operations for preventative and corrective maintenance actions with spare parts management.

2. A novel formulation for solving a MRSP with the objective of minmising revenue loss from WT downtime which allows maintenance jobs to be split across multiple shifts.

3. A novel solution approach for solving the MRSP based on an *Adaptive Large Neighbourhood Search* (ALNS) metaheuristic and a matheuristic, which outperforms a commercial solver in terms of solution quality and time for some problem instances.

4. An evaluation of the framework on a whole year's worth of historical data from an OWF, the longest evaluation period considered so far to the best of the author's knowledge, indicating potential savings of 66%.

The remainder of the article is structured as follows: a review of tactical and operational OWF planning models is presented in §2. The combined framework is introduced in §3 which is followed by the model formulation and solution approach for the long-term job allocation in §4. The formulation for the MRSP is then introduced in §5 and the ALNS solution method is described in §6. This is followed by the results of computational experiments to evaluate the performance of the formulations and their respective solution approaches in §7. Finally, the article comes to a close with a conclusion in §8.

2. Literature Review

A summary of the literature for maintenance routing and scheduling of OWFs on the tactical and operational scale is provided in Table 1. The objective of each approach; the constraints they consider;

the perspectives they are considered from; the uncertainties; and their validation is compared with each other and the method proposed in this research.

Table 1: Summary of research in tactical and operational maintenance scheduling and routing.

	This article	[11]	[13]	[2]	[20]	[7]	[10]
Tactical	✓	✓	✗	✗	✗	✗	✗
Operational	✓	✓	✓	✓	✓	✓	✓
Time Scale	1 year	30 days	7 days	1 day	1 day	1 day	7 days
Problem Considered	MRSP	MRP &MSP	MRP & MSP	MRSP	MRSP	MRP	MRP
Model	MILP	MILP	-	-	-	MIP	MILP
Perspective	Operator	Operator	MS	Operator	Operator	MS	MS
Validation	Operational Data	Simulation	Simulation	Simulation	Operational Data	Operational Data	Simulation
Objective	Min revenue loss	Min cost	Min cost	Min cost	Min cost	Min cost	Min cost
Vessel Routing	✓	✓	✓	✓	✓	✓	✓
Maintenance Scheduling	✓	✓	✓	✓	✓	✗	✗
Spare Parts Management	✓	✗	✗	✗	✗	✗	✗
Interrupted Jobs	✓	✗	✗	✗	✗	✗	✗

The routing and scheduling of maintenance tasks can be separated into the *Maintenance Routing Problem* (MRP) and *Maintenance Scheduling Problem* (MSP). Several studies have grouped them into the MRSP, whereby the route of the vessel determines the schedule. The MRP can be considered a variation of the *Vehicle-Routing Problem with Pickup and Delivery* (VRPDP).

Stålhane *et al.* [21] formulate the deterministic MRP as an arc-flow model and a path-flow model which are solved by a branch-and-cut algorithm and a heuristic, respectively. The objective is to minimise the costs of the routes taken by the vessel on a daily basis. Dawid *et al.* [6] formulate the MRP as a *Mixed-Integer Linear Programming* (MILP) model where the objective is to minimise daily routing costs and is solved with a heuristic. This is later generalised by Dawid *et al.* [7] to incorporate uncertainties in transfer times, failure rates and maintenance duration which they validated on historical data from a wind farm. Irawan *et al.* [10] formulate the problem, for multiple wind farms and O&M bases as an MILP which is solved with CPLEX to minimise the costs of vessel routes. Their model is validated on simulated data across one week of operations. Schrotenboer *et al.* [17] consider vessel routing in combination with technician sharing, from the viewpoint of a maintenance service provider in the technician allocation and routing problem as a MILP which is solved with an ALNS metaheuristic. Their objective is also to minimise the cost of routes. Neves-Moreira *et al.* [14] consider both the MRP and repair kit problem as a two-stage stochastic MILP model. They consider routing and maintaining optimal stock levels of spare parts considering weather and failure rate uncertainties and it is solved using a matheuristic. Irawan *et al.* [9] propose a simulation-based MILP to solve the Stochastic MRP, where uncertainties in travel time, transfer time, and maintenance duration are considered. Their model is solved by a large neighbourhood search metaheuristic. Except for [7], all of these models have been validated on simulated case studies rather than operational data.

Dai *et al.* [5] formulated the MRSP as a MILP for minimising the costs for a single wind farm and O&M base for single day of maintenance and it is solved with Xpress. Raknes *et al.* [15] expands the aforementioned MILP formulation to multiple wind farms and vessels, and solved it with a heuristic. Stock-Williams & Swamy [20] propose a simulation based formulation and genetic algorithm to minimise the cost of daily scheduling and vessel routing. Schrotenboer *et al.* [16] propose a branch-and-cut algorithm for minimising the cost of daily routing, with consideration of different technician availabilities to solve a variation of the vehicle pickup and delivery problem. Lazakis & Khan [13] solve the MRP and MSP using a simulation based heuristic to minimise the cost of routing and scheduling maintenance. Allal *et al.* [2] also employ a simulation based meta-heuristic to minimise the cost of daily maintenance planning and routing whilst aiming to keep turbine availability high. They employ an ant colony optimisation metaheuristic to solve the problem. Schrotenboer *et al.* [18] incorporates, weather; failure rate; maintenance duration; and technician requirement uncertainties in the stochastic maintenance fleet transportation problem for offshore wind farms. They take the perspective of a maintenance service provider to minimise their costs of maintenance whilst providing minimum service levels, with multiple fleets and across multiple wind farms. It is formulated as a two-stage MILP model and solved using sample average approximation. Only Stock-Williams & Swamy [20] has validated their model on operational data.

Irawan *et al.* [11] is the only research, to the best of the authors knowledge, to consider a combined tactical and operational problem. In this work they present a framework for scheduling and routing at a wind farm for a 30-day period. It is formulated as a MILP and is decomposed into an MSP which is solved exactly on the long term. On each day of the schedule the list of maintenance jobs is updated with corrective maintenance actions and an MRP is solved with a metaheuristic. If all maintenance actions can't be routed on a given day the MSP is resolved. The objective is to minimise the total cost of maintenance. This is the most detailed framework to date that can be applied to regular maintenance actions.

This research builds on the models discussed above by capturing a more complete set of constraints and operating practices by taking the perspective of a wind farm operator. It aims to provide actionable insights that can be used in practice which has previously only been done by Irawan *et al.* [11]. None of the models discussed above consider minimisation of revenue losses as the objective, only Stock-Williams & Swamy [20] accounts for multiple jobs on the same turbine occurring at the same time, and only Raknes *et al.* [15] considers maintenance actions that take multiple periods. For models that solve both the MRP and MSP, these are solved separately. When this approach is adopted the quality of the route impacts the quality of the schedule and *vice versa*, or the scheduling is based on the routing which does not consider how the scheduling can affect profitability. The work presented here addresses these gaps by presenting a framework for planning maintenance over both the tactical and operational timescales. A novel formulation for solving the MRSP, which addresses the gaps in constraints considered by the previous formulations, with the objective of minimising revenue losses from turbine downtime is presented with a novel ALNS solution method. Finally the method is validated against historical data from an operational OWF, solved for a full year of operation.

3. Maintenance Planning Framework

The current process undertaken for planning maintenance at the OWF considered as a case-study in this article is as follows. First, the maintenance team are provided with a plan for the maintenance jobs over the coming year. This plan outlines known maintenance jobs (e.g. annual servicing and inspections) that will need to be performed and when in the year they should be planned to take place. Tasks are allocated to times of the year when it is expected they will most likely be performed. Then, on a daily basis, corrective and routine tasks are planned. During an evening maintenance planning meeting, a short-term weather forecast for the next few days is evaluated with a list of known jobs that need to be performed on the wind farm. If the weather is expected to be good enough to perform the maintenance, then the operators plan to do the jobs at the wind farm. If it is not, then the operators plan not to perform any maintenance in the short term. Then in the following morning another maintenance planning meeting is held with updated weather forecasts and any jobs that have arrived overnight. The weather is reassessed and if it is safe enough, the maintenance teams go to site and perform the planned maintenance, if not then no maintenance is performed. This process is then repeated daily. Revenue loss associated with planning maintenance is not directly considered but by trying to plan maintenance actions for periods where the weather is calm, therefore wind speed is low, energy loss from WT being offline is also low. This also gives maintenance the most likely chance of succeeding. Any corrective maintenance actions are performed as soon as possible. Therefore, there is scope in this process to both minimise revenue losses from planned maintenance and costs of performing maintenance in a more systematic manner.

The framework for producing maintenance schedules which minimise cost across tactical and operational timescales is illustrated graphically in Figure 1. The problem of simultaneously solving both the tactical and operational timescales is divided into a *Long-Term* (LT) job allocation problem and a *Shorter-Term* (ST) MRSP. Starting with a set of known maintenance jobs that need to be performed at the wind farm and LT forecasts of weather and electricity prices for each time $t \in \mathcal{T}$. The set $\mathcal{T} = \{0, \ldots, T^{\max}\}$ contains the indices of the discrete long term decision periods, for example the months in a year. Each job is then allocated to a long term planning period $t \in \mathcal{T}$ such that the expected revenue losses and costs for performing these maintenance actions are minimised, step 1 in Figure 1. The minimisation is based on expected time available for maintenance in each planning period $t \in \mathcal{T}$, expected revenue losses, maintenance deadlines and routing costs. A conservative approach in the allocation is taken to allow enough time in each planning period $t \in \mathcal{T}$ for updated weather forecasts and new maintenance jobs.

Once an initial LT plan has been produced, an updated list of maintenance jobs, such as corrective maintenance and new ST forecasts of wind farm accessibility and revenue generation are acquired by the maintenance team, step 2 in Figure 1. The long-term job allocation problem is then resolved based on the updated information, step 3 in Figure 1, as a multi-objective optimisation problem,

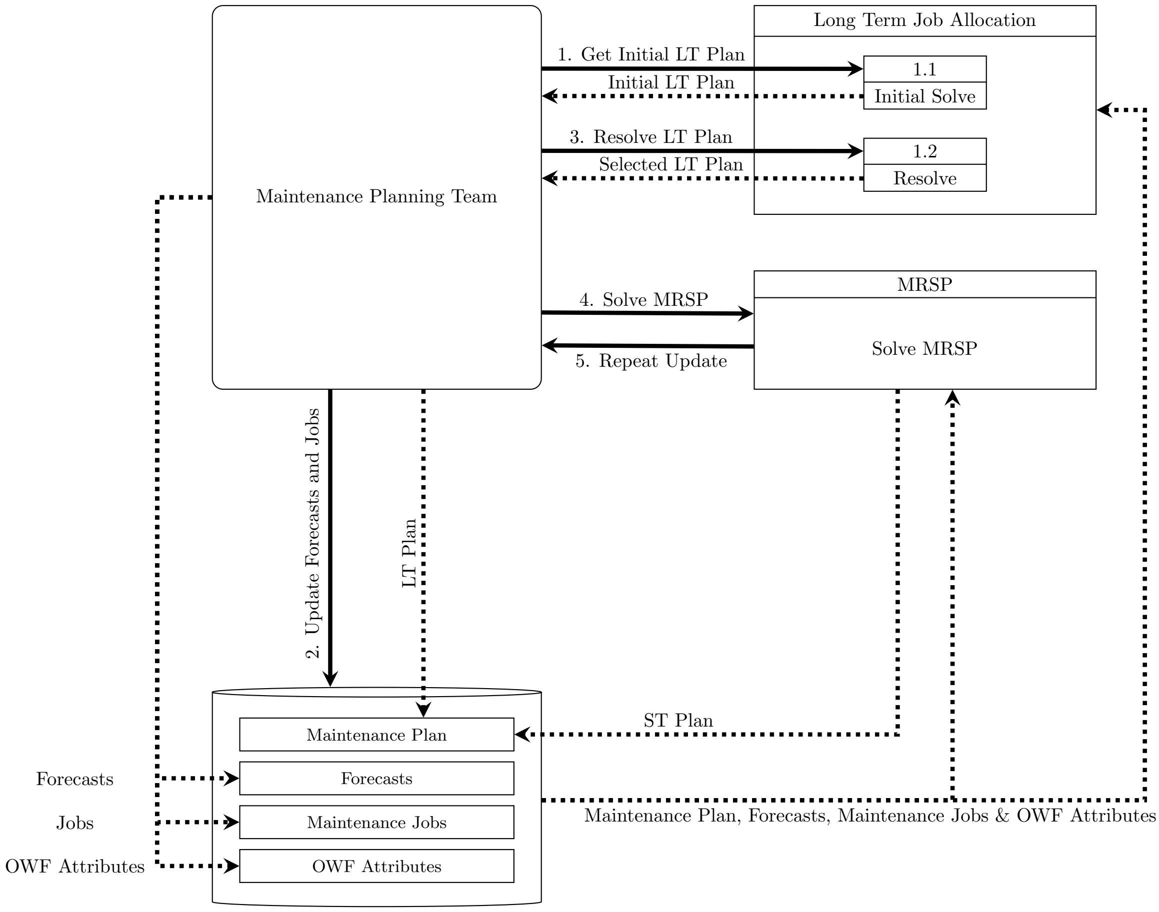

Figure 1: Proposed framework for tactical and operational maintenance planning for an OWF. Dotted lines indicate a flow of information and solid lines indicate the process flow.

where the objectives are to minimise the disruption to the original long-term plan and total cost of job allocation. Multiple solutions are provided and the maintenance team select their preferred option. Then for each index $t \in \mathcal{T}$ the planning period is discretised into smaller planning periods denoted by $\hat{\mathcal{T}} = \{0, \ldots, \hat{T}^{\max}\}$. Each set $\hat{\mathcal{T}}$ contains the indices of the short-term planning periods, typically in days or hours. This is illustrated graphically in Figure 2. The MRSP is then solved across each $t \in \hat{\mathcal{T}}$, step 4 in Figure 1. This process continues on a daily or per shift basis.

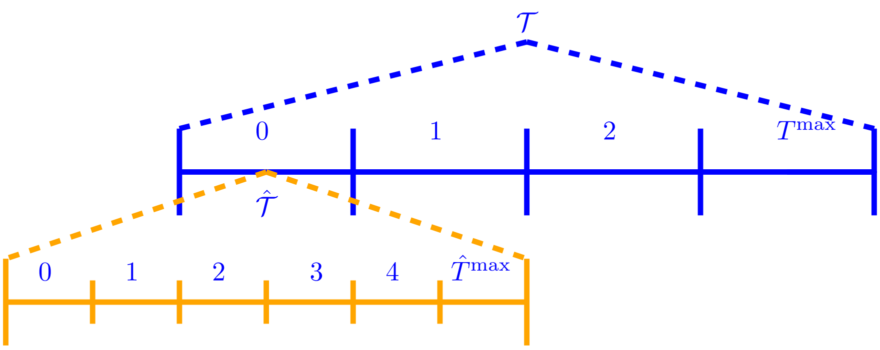

Figure 2: A graphical illustration of the discretisation of long-term planing periods into short-term planning periods.

In the following sections all variables related to the long-term job allocation problem will have no accent (e.g., x_{wjt}); variables related to scheduling will have a circumflex accent (e.g.,$\hat{x}_{wjt}$); and variables related to the routing will have a tilde accent (e.g., $\tilde{x}_{ijd}$). The parameters and variables used in the following sections are summarised in Tables A.5–A.7 in the Appendix.

4. Long-Term Job Allocation

In this section the formulation of the MILP model for the long-term job allocation problem is described. The decision variables and constraint sets associated with assigning jobs to planning periods $t \in \mathcal{T}$ are presented in §4.1. The decision variables and constraint sets used to model the management of spare parts are presented in §4.2. In §4.3 the decision variables and constraint sets for modelling WT downtime are presented and the objective function of the MILP is described. Finally, the decision variables; constraint sets and additions to the objective function necessary for modelling the multi-objective resolve, as described in stage 3 of the framework in §3, are presented in §4.4.

4.1. Job Assignment

Let $\mathcal{W} = \{0, \ldots, n\}$ denote the set of indices for the WTs at the OWF and the set of jobs to be performed at a turbine $w \in \mathcal{W}$ be denoted by $j \in \mathcal{J}_w$. Moreover, let the travel time from turbine $w \in \mathcal{W}$ to turbine $w' \in \mathcal{W}$ be denoted by $\tau_{ww'}$ with an associated cost of $c_{ww'}$, whilst the travel time from O&M base to turbine is denoted by $\bar{\tau}_w$. Let δ_{wj} denote the time required to complete job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$ which has a deadline by which it must be performed denoted by l_{wj} and an earliest start date denoted by l^*_{wj}. Jobs $j \in \mathcal{J}^*_w$ lead to a failure of the turbine if their deadline is exceeded and jobs $j \in \mathcal{J}'_w$ do not if their deadline is exceeded. The set of jobs is therefore expressed as $\mathcal{J}_w = \mathcal{J}^*_w \cup \mathcal{J}'_w$. Jobs that do not lead to a failure are typically regular inspections or annual services. These jobs must be performed by a specific deadline for contractual and insurance purposes, therefore there is a hard deadline by which they must be performed. For jobs that lead to a failure, the penalty cost for late maintenance is equal to the revenue loss for each time where the turbine is offline. Some ordered job pairs $(j, j') \in \mathcal{R}_w$ where $\mathcal{R}_w$ is the set of all job pairs that must be performed at regular intervals denoted by $\bar{l}_{wjj'}$, i.e. if job j is scheduled at time t, then job j' can only be scheduled after time $t + \bar{l}_{wjj'}$. Jobs also require technicians to perform them. Therefore, let the number of technicians required for a job $j \in \mathcal{J}_w$ on turbine $w \in \mathcal{W}$ be denoted by ρ_{wj} and the number available in $t \in \mathcal{T}$ be denoted by ρ'_t. The amount of time available, due to weather constraints, to perform maintenance in each $t \in \mathcal{T}$ is denoted by e_t. To assign each job to a time period the following decision variables are introduced

$$x_{wjt} = \begin{cases} 1 & \text{if job } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ is scheduled in period } t \in \mathcal{T} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$y_{ww't} = \begin{cases} 1 & \text{if turbines } w, w' \in \mathcal{W} \text{ are both assigned to period } t \in \mathcal{T} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$z_t = \text{the assignment cost of period } t \in \mathcal{T},$$

and the following constraints

$$\sum_{t \in \mathcal{T}} x_{wjt} = 1 \quad \forall w \in \mathcal{W},\ j \in \mathcal{J}_w, \tag{1}$$

$$y_{w'wt} \geq x_{w'j't} + x_{wjt} - 1 \quad \forall t \in \mathcal{T},\ w, w' \in \mathcal{W},\ j, j' \in \mathcal{J}_w, \tag{2}$$

$$z_t \geq c_{ww'} y_{ww't} + c_{w'w} y_{w'wt} \quad \forall t \in \mathcal{T},\ w, w' \in \mathcal{W}, \tag{3}$$

$$\sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} x_{wjt} \delta_{wj} \rho_{wj} + \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} 2 x_{wjt} \bar{\tau}_w \leq e_t \rho'_t \quad \forall t \in \mathcal{T}, \tag{4}$$

$$\sum_{t \in \mathcal{T}} t x_{wjt} \geq l^*_{wj} \quad \forall w \in \mathcal{W},\ j \in \mathcal{J}_w, \tag{5}$$

$$\sum_{t \in \mathcal{T}} t x_{wjt} \leq l_{wj} \quad \forall w \in \mathcal{W},\ j \in \mathcal{J}'{}_w, \tag{6}$$

$$\sum_{t \in \mathcal{T}} t x_{wjt} + \bar{l}_{wjj'} \leq \sum_{t \in \mathcal{T}} t x_{wj't} \quad \forall w \in \mathcal{W},\ (j, j') \in \mathcal{R}_w. \tag{7}$$

Constraint set (1) ensures that every job is assigned to exactly one time period. Constraint set (2) ensures that if two jobs are assigned to a planning period, they are also assigned to a turbine pairing. Constraint set (3) determines a worst-case travel cost between the pair of turbines. The time available to do maintenance in each $t \in \mathcal{T}$ is limited by the weather accessibility, working hours of the maintenance teams and the number of technicians available. Therefore, constraint set (4) is introduced to ensure that the total time spent on maintenance jobs and on travel to and from the farm does not exceed the available time. In this scenario a worst case travel time is assumed where the vessel must travel to and from the farm independently to perform each job in the planning period, without considering intra-farm travel. The time required and the time available to perform maintenance are considered as 'technician time', which is a function of both time and technicians available. In this way the ability to schedule multiple jobs in parallel on a given day is captured. Constraint sets (5)–(6) ensure that all jobs are performed after their earliest start date and all non-failure jobs are performed before their deadline. Constraint set (7) ensures that jobs which happen at regular intervals are not scheduled too close together.

4.2. Spare Parts

A job cannot be performed if the spare parts are not in stock, therefore the following parameters are introduced. Let α_{wjf} denote the quantity of part $f \in \mathcal{F}$ required to perform a job $j \in \mathcal{J}_w$. Let the lead time to acquire part $f \in \mathcal{F}$ be denoted by λ_f and the initial stock in time $t \in \mathcal{T}$ be denoted by $\bar{\alpha}_{ft}$. The holding cost is denoted by c^{inv}. To track the spare parts inventory and required orders, the following decision variables are introduced

$$s_{ft} = \text{the stock of part } f \in \mathcal{F} \text{ at the end of period } t \in \mathcal{T},$$
$$r_{ft} = \text{the quantity of part } f \in \mathcal{F} \text{ ordered in period } t \in \mathcal{T},$$

and the following constraints

$$s_{ft} = s_{ft-1} + \bar{\alpha}_{ft} - \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} a_{wjf} x_{wjt} \quad \forall t \in \mathcal{T} \setminus \{0\}, f \in \mathcal{F}, : t - \lambda_f < 0, \tag{8}$$

$$s_{ft} = s_{ft-1} + \bar{\alpha}_{ft} + r_{f(t-\lambda_p)} - \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} a_{wjf} x_{wjt} \quad \forall t \in \mathcal{T} \setminus \{0\}, f \in \mathcal{F}, : t - \lambda_f \geq 0, \tag{9}$$

$$r_{ft}, s_{ft} \geq 0 \quad \forall f \in \mathcal{F}, t \in \mathcal{T}. \tag{10}$$

Constraint sets (8) and (9) set the stock of spare parts in each time period. Constraint set (10) ensures that work can only be planned if the spare parts are available.

4.3. Objective Function

Jobs $j \in \mathcal{J}_w^*$ should be assigned before their deadline and after their earliest start date. If they are not, then they are subject to a revenue loss which is dependent on how long the job was delayed. Therefore, the following parameters are introduced. The cost to perform a corrective job $j \in \mathcal{J}_w^*$ after failure is denoted by c_{wj}^*, whilst the cost to perform a preventative job given no failure is c'_{wj}. The cumulative revenue loss from 0 up to time $t \in \mathcal{T}$ is given by r'_t. To track the delayed jobs and revenue losses the following decision variable is introduced

$$o_{wjt} = \begin{cases} 1 & \text{if job } j \in \mathcal{J}_w^* \text{ at turbine } w \in \mathcal{W} \text{ is delayed and assigned to period } t \in \mathcal{T} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

and the following constraints

$$t x_{wjt} \geq l_{wj} - T^{\max}(1 - o_{wjt}) \quad \forall t \in \mathcal{T}, w \in \mathcal{W}, \ j \in \mathcal{J}_w^*, \tag{11}$$

$$t x_{wjt} \leq l_{wj} + (T^{\max} - l_{wj}) o_{wjt} \quad \forall t \in \mathcal{T}, w \in \mathcal{W}, \ j \in \mathcal{J}_w^*, \tag{12}$$

$$\sum_{t \in \mathcal{T}} o_{wjt} \leq 1 \quad \forall w \in \mathcal{W}, \ j \in \mathcal{J}_w^*, \tag{13}$$

to determine the time at which a maintenance $j \in \mathcal{J}_w^*$ is performed if it is performed late.

The revenue loss from performing a job $j \in \mathcal{J}_w$ in a planning period $t \in \mathcal{T}$ is dependent on the total duration of jobs assigned to that planning period. If there is only a single job assigned to a planning period $t \in \mathcal{T}$, then it can be assigned to the lowest revenue time in the short term planning period $t \in \hat{\mathcal{T}}$. As the total duration of jobs assigned to the planning period increases the cost of assignment to that period increases. Therefore, the following decision variable is introduced to determine the utilisation of each planning period

$$u_{wt} = \text{the utilisation of turbine } w \in \mathcal{W} \text{ in period } t \in \mathcal{T},$$

and the following constraint set

$$u_{wt} = \left(\sum_{j \in \mathcal{J}_w} x_{wjt} \delta_{wj} \right) / e_t \quad \forall w \in \mathcal{W}, t \in \mathcal{T}, \tag{14}$$

determines the total duration of all jobs at each turbine divided by the time available to perform maintenance in each time period $t \in \mathcal{T}$, i.e. the turbine utilisation.

The objective is to minimise the sum of the revenue losses from performing maintenance; the costs of travel; the costs of performing maintenance and the inventory holding costs. The objective function is therefore defined as

$$\min \; S_0 = \tag{15}$$

$$\sum_{w \in \mathcal{W}} \sum_{t \in \mathcal{T}} U_{wt}(u_{wt}) + \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w^*} \left(\sum_{t \in \mathcal{T}} (r'_t o_{wjt}) - r'_{l_{(wj)}} \sum_{t \in \mathcal{T}} o_{wjt} \right) \tag{16}$$

$$+ \sum_{t \in \mathcal{T}} z_t \tag{17}$$

$$+ \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}'_w} c'_{wj} + \sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w^*} \left(\sum_{t \in \mathcal{T}} \left(o_{wjt} c^*_{wj} \right) + c'_{wj} \left(1 - \sum_{t \in \mathcal{T}} o_{wjt} \right) \right) \tag{18}$$

$$+ \sum_{f \in \mathcal{F}} \sum_{t \in \mathcal{T}} s_{pt} c^{\text{inv}}. \tag{19}$$

Expression (16) considers both the revenue losses from WT downtime due to maintenance being performed, plus any revenue losses from WT downtime due to failures at the turbine. The revenue losses from performing maintenance in a planning period $t \in \mathcal{T}$ denoted by U_{wt} is a function of the total utilisation of the turbine in that planning period u_{wt}. The revenue losses from failure at a turbine are the sum of revenue losses between the deadline for a job and the time at when maintenance has been assigned, only for jobs that are performed late. The assignment cost in (17) is the sum of all assignment costs for all $t \in \mathcal{T}$. The maintenance cost in (18) is made up of preventative maintenance costs for all jobs that don't lead to a failure, plus the preventative costs for failure jobs that were not scheduled late, plus the corrective maintenance costs for jobs that were performed late. Finally, the inventory holding cost in (19) accounts for the storage cost of the spare parts.

4.4. Multi-objective Resolve

As indicated by step 3 in the framework presented in §3, when new maintenance jobs arise, the original tactical maintenance plan must be adapted while simultaneously minimising both the total cost, including revenue losses, and changes to the original schedule. Let the set of turbines that originally had jobs scheduled at them be denoted by $w \in \bar{\mathcal{W}}$. Let the set of jobs, from the original list, that have not been completed by the point of the resolve be denoted by $j \in \bar{\mathcal{J}}_w$. The original assignment variables are denoted by $\bar{x}_{wjt}$. To track the changes in assignment period for each job the following decision variable is introduced,

$$v_{wj} = \text{the change in assignment time for } j \in \bar{\mathcal{J}}_w \text{ at turbine } w \in \bar{\mathcal{W}},$$

and the following constraint set

$$v_{wj} = \left| \sum_{t \in \mathcal{T}} t x_{wjt} - \sum_{t \in \mathcal{T}} t \bar{x}_{wjt} \right| \quad \forall w \in \bar{\mathcal{W}}, j \in \bar{\mathcal{J}}_w. \tag{20}$$

Constraint set (20) tracks how far forwards or backwards in time each job $j \in \bar{\mathcal{J}}_w$ is moved in the new schedule. A linearisation of this constraint set is performed. All the constraints, 1 − 14, and variables, $x_{wjt}, y_{ikt}, z_t, s_{ft}, r_{ft}, o_{wjt}, u_{wjt}$, described above are included in the resolve model. The objective functions are to simultaneously minimise S_0 in (15) and

$$S_1 = \sum_{w \in \bar{\mathcal{W}}} \sum_{j \in \bar{\mathcal{J}}_w} v_{wj}. \tag{21}$$

In other words, the objective is to now minimise both cost and the time magnitude of changes to the long-term plan.

5. Maintenance Routing and Scheduling Problem

Once jobs are allocated to a long-term planning period $t \in \mathcal{T}$ (e.g., one month), exact schedules and routes for the discrete planning periods $t \in \hat{\mathcal{T}}$ must be determined. This is done by solving the MRSP and the output is a schedule of when maintenance jobs should be performed and the routes that the maintenance vessels should take. The quality of a route can impact the quality of a schedule and vice-versa. Therefore they must be solved together to guarantee global optimality.

In this section the formulation of the MILP model for the MRSP is described. In §5.1 the decision variables and constraint sets associated with assigning jobs to planning periods $t \in \hat{\mathcal{T}}$ are presented. The decision variables and constraint sets used to model the technicians and spare parts limits are presented in §5.2. In §5.3 the decision variables and constraint sets for modelling WT downtime are presented and the objective function of the MILP is described in §5.4. The decision variables, constraint sets and additions to the objective function necessary for modelling the routing of the maintenance vessel are presented in §5.5.

Whilst solved together, it is convenient to separate the formulation description into sets, parameters and constraints that affect the scheduling and those that affect the routing. Common sets and parameters with the long-term job allocation problem keep the same notation, described above in §4.

5.1. Job Scheduling

Consider a single long-term planning period $t \in \mathcal{T}$ and let the index of discrete time periods be denoted by $t \in \hat{\mathcal{T}}$, practically these can be considered as hours of the day for example. Let the set of discrete maintenance shifts be denoted by $d \in \hat{\mathcal{D}}$, practically these can be considered as days or consecutive 12-hour shifts. Let $t \in \hat{\mathcal{T}}_d$ denote the set of indices in each discrete maintenance shift and the start and end index for work on each shift as $\hat{T}_d^{\min}$ and $\hat{T}_d^{\max}$, respectively, it is assumed that $\hat{\mathcal{T}} = \bigcup_{d \in \hat{\mathcal{D}}} \hat{\mathcal{T}}_d$. The length of each shift and its start and end times are determined by the wind farm operating practices. Each job can only be performed when the turbine is accessible for maintenance and each job has different accessibility criteria determined by technician working hours and weather conditions, this is described by the binary indicator

$$\hat{h}_{wjt} = \begin{cases} 1 & \text{if job } j \in \mathcal{J}_w \text{ can be performed at turbine } w \in \mathcal{W} \text{ at time } t \in \hat{\mathcal{T}} \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

The accessibility of each job is also determined by its area within the turbine, e.g. nacelle or tower. Let the set of areas in a turbine be denoted by $b \in \hat{\mathcal{B}}$, the location of each job within a turbine is described by the binary indicator

$$\hat{\beta}_{wjb} = \begin{cases} 1 & \text{if job } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ is performed in turbine area } b \in \hat{\mathcal{B}} \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, let the maximum number of jobs that can be performed in any of the turbine areas at once be denoted by $\hat{B}_b^{\max}$. To consider the assignment of jobs to time periods the following decision variables are introduced

$$\hat{x}_{wjt} = \begin{cases} 1 & \text{if maintenance for job } j \in \mathcal{J}_w \text{ is occurring at turbine } w \in \mathcal{W} \text{ at time } t \in \hat{\mathcal{T}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{y}_{wjt} = \begin{cases} 1 & \text{if maintenance for job } j \in \mathcal{J}_w \text{ is starting at turbine } w \in \mathcal{W} \text{ at time } t \in \hat{\mathcal{T}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{u}_{wjt} = \begin{cases} 1 & \text{if maintenance for job } j \in \mathcal{J}_w \text{ ends at turbine } w \in \mathcal{W} \text{ at the end of time } t \in \hat{\mathcal{T}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{q}_{wd} = \begin{cases} 1 & \text{if turbine } w \in \mathcal{W} \text{ is visited during maintenance shift } d \in \hat{\mathcal{D}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

and the following constraints

$$\sum_{t \in \hat{\mathcal{T}}} \hat{y}_{wjt} \geq 1 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{22}$$

$$\hat{y}_{wj0} \geq \hat{x}_{wj0} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{23}$$

$$\hat{y}_{wjt} \geq \hat{x}_{wjt} - \hat{x}_{wj(t-1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{24}$$

$$\hat{y}_{wjt} \leq \hat{x}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{25}$$

$$\hat{u}_{wj\hat{T}^{\max}} \geq \hat{x}_{wj\hat{T}^{\max}} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{26}$$

$$\hat{u}_{wjt} \geq \hat{x}_{wjt} - \hat{x}_{wj(t+1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{\hat{T}^{\max}\}, \tag{27}$$

$$\hat{u}_{wjt} \leq \hat{x}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{28}$$

$$\sum_{t \in \hat{\mathcal{T}}} \hat{y}_{wjt} = \sum_{t \in \hat{\mathcal{T}}} \hat{u}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{29}$$

$$\sum_{t \in \hat{\mathcal{T}}} \hat{x}_{wjt} = \delta_{wj} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{30}$$

$$\sum_{j \in \mathcal{J}_w} \sum_{t \in \hat{\mathcal{T}}_d} \hat{y}_{wjt} \leq |\mathcal{J}_w| q_{wd} \quad \forall w \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{31}$$

$$\hat{x}_{wjt} \leq \hat{h}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{32}$$

$$\hat{y}_{wjt} \leq \hat{h}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{33}$$

$$\hat{y}_{wjt} = 0 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, d \in \hat{\mathcal{D}}, t \in \hat{\mathcal{T}}_d : t \leq \hat{T}_d^{\min} + \bar{\tau}_w, \tag{34}$$

$$\hat{u}_{wjt} = 0 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, d \in \hat{\mathcal{D}}, t \in \hat{\mathcal{T}}_d : t \geq \hat{T}_d^{\max} + \bar{\tau}_w, \tag{35}$$

$$\sum_{j \in \mathcal{J}_w} \hat{x}_{wjt} \hat{\beta}_{wjb} \leq \hat{B}_b^{\max} \quad \forall w \in \mathcal{W}, d \in \hat{\mathcal{D}}, t \in \hat{\mathcal{T}}, b \in \hat{\mathcal{B}}. \tag{36}$$

Constraint set (22) ensures that maintenance is started for each job at least once. Constraint sets (23)–(25) and (26)–(28) set the start and end times of maintenance, respectively and ensure that maintenance is only performed once started and before finished. Constraint set (29) ensures that the number of times a job is started is equal to the number of times it finishes. Constraint set (30) ensures that the time spent performing maintenance for each job is equal to the required duration to complete the job. Constraint set (31) tracks which turbines are visited in each shift. Constraint sets (32) and (33) ensure that maintenance cannot be performed or started when the turbine is not accessible. Constraint set (34) ensures that maintenance cannot be scheduled before the turbine can be reached from the base in each shift. Constraint set (35) ensures that maintenance is not scheduled during times which do not permit travel back to the base at the end of a shift. Constraint set (36) ensures that the number of jobs occurring in each area of the turbine at any one turbine is less than the limit for that area.

5.2. Scheduling - Technician and Spare Parts Limits

Each job requires specialised technicians and spare parts to complete it. Let technician type be denoted by $p \in \hat{\mathcal{P}}$, and the number of technicians of each type required for a job $j \in \mathcal{J}_w$ on turbine

$w \in \mathcal{W}$ be denoted by $\hat{\rho}_{wjp}$ and their availability in $t \in \hat{\mathcal{T}}$ be denoted by $\hat{\rho}'_{pt}$. Let the weight of spare parts required for a job $j \in \mathcal{J}_w$ on turbine $w \in \mathcal{W}$ be denoted by $\hat{\omega}_{wj}$ and the vessel spare parts capacity limit be denoted by $\hat{\omega}^{\max}$. It is assumed that because of the spare parts planning included in the long-term job assignment problem, all spare parts needed for maintenance are available. The number of technicians needed for maintenance at a turbine during any maintenance shift must be known. If jobs are performed consecutively, only one set of technicians may be needed to perform both jobs, whereas if they are performed in parallel, multiple sets of technicians are needed. To capture the number of technicians needed for maintenance at a turbine, and spare parts and technician limits the following decision variable is introduced,

$$\hat{s}_{wd} = \text{the number of technicians needed at turbine } w \in \mathcal{W} \text{ for shift } d \in \hat{D},$$

together with the constraints

$$\sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} \hat{x}_{wjt} \hat{\rho}_{wjp} \leq \hat{\rho}'_{pt} \quad \forall p \in \hat{\mathcal{P}}, d \in \hat{\mathcal{D}}, t \in \hat{\mathcal{T}}_d \tag{37}$$

$$\sum_{w \in \mathcal{W}} \sum_{j \in \mathcal{J}_w} \sum_{t \in \hat{\mathcal{T}}_d} \hat{y}_{wjt} \hat{\omega}_{wj} \leq \hat{\omega}^{\max} \quad \forall d \in \hat{\mathcal{D}} \tag{38}$$

$$\sum_{j \in \mathcal{J}_w} \sum_{p \in \hat{\mathcal{P}}} \hat{x}_{wjt} \hat{\rho}_{wjp} \leq \hat{s}_{wd} \quad \forall w \in \mathcal{W}, d \in \hat{D}, t \in \hat{\mathcal{T}}_d. \tag{39}$$

Constraint set (37) ensures crew assigned for maintenance do not exceed the crew available. Constraint set (38) ensures the parts being used for maintenance in a shift do not exceed the vessel capacity limit. Constraint set (39) tracks the total number of technicians needed per turbine in a shift and is used for the vessel routing constraints.

5.3. Scheduling - Tracking Downtime

Once a job is started the turbine is offline and unable to generate revenue until it is finished. If the maintenance for a job requires multiple starts the revenue loss will be between the very first start time of that job and the very last end time of that job. If multiple jobs need to happen on the same turbine, it will be cost saving to perform these jobs in parallel as the turbine will be offline anyway. As for the long-term job allocation problem there are penalties for performing maintenance late, which are a function of how late the maintenance is and whether a corrective or preventative replacement are required. To track the periods where the turbine is offline, either due to failure or maintenance, and to track how delayed each job is, the following decision variables are introduced

$$\hat{k}^1_{wjt} = \begin{cases} 1 & \text{from time } t \in \hat{\mathcal{T}} \text{ once job } j \in \mathcal{J}_w \text{ has started at turbine } w \in \mathcal{W} \text{ for the first time and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{k}^2_{wjt} = \begin{cases} 1 & \text{from time } t \in \hat{\mathcal{T}} \text{ once job } j \in \mathcal{J}_w \text{ has finished at turbine } w \in \mathcal{W} \text{ for the last time and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{k}^3_{wjt} = \begin{cases} 1 & \text{in time } t \in \hat{\mathcal{T}} \text{ when turbine } w \in \mathcal{W} \text{ is offline because of job } j \in \mathcal{J}_w \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

$$\hat{v}^1_{wjt} = \begin{cases} 1 & \text{in time } t \in \hat{\mathcal{T}} \text{ after first start for job } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{v}^2_{wjt} = \begin{cases} 1 & \text{in time } t \in \hat{\mathcal{T}} \text{ after last finish for job } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{m}_{wt} = \begin{cases} 1 & \text{if turbine } w \in \mathcal{W} \text{ if offline in time } t \in \hat{\mathcal{T}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{z}_{wjt} = \begin{cases} 1 & \text{for time } t \in \hat{\mathcal{T}} \text{ between a job's } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ deadline and start if late and} \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{o}_{wj} = \begin{cases} 1 & \text{if job } j \in \mathcal{J}_w \text{ at turbine } w \in \mathcal{W} \text{ is scheduled late and} \\ 0 & \text{otherwise,} \end{cases}$$

together with the constraints

$$\hat{k}^1_{wj0} \leq \hat{y}_{wj0} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{40}$$

$$\hat{k}^1_{wjt} \geq \hat{y}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{41}$$

$$\hat{k}^1_{wjt} \geq \hat{k}^1_{wjt-1} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{42}$$

$$\hat{k}^1_{wjt} \leq \hat{y}_{wjt} + \hat{k}^1_{wjt-1} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{43}$$

$$\sum_{t \in \hat{\mathcal{T}}} \hat{v}^1_{wjt} = 1 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{44}$$

$$\hat{v}^1_{wj0} \geq \hat{k}^1_{wj0} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{45}$$

$$\hat{v}^1_{wjt} \leq \hat{k}^1_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{46}$$

$$\hat{v}^1_{wjt} \leq 1 - \hat{k}^1_{wj(t-1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{47}$$

$$\hat{v}^1_{wjt} \geq \hat{k}^1_{wjt} - \hat{k}^1_{wj(t-1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{48}$$

$$\sum_{t \in \hat{\mathcal{T}}} \hat{v}^2_{wjt} = 1 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{49}$$

$$t\hat{v}^2_{wjt} \geq t\hat{u}_{wjt} - \sum_{t' \in \hat{\mathcal{T}}: t' > t} t'\hat{u}_{wjt'} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{50}$$

$$\hat{v}^2_{wj\hat{\mathcal{T}}^{\max}} = \hat{u}_{wj\hat{\mathcal{T}}^{\max}} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{51}$$

$$\hat{k}^2_{wjt} \geq \hat{v}^2_{wj(t-1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{52}$$

$$\hat{k}^2_{wjt} \geq \hat{k}^2_{wj(t-1)} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} \setminus \{0\}, \tag{53}$$

$$\hat{k}^2_{wjt} \leq \sum_{t' \in \hat{\mathcal{T}}: t' < t} \hat{v}^2_{wjt'} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{54}$$

$$\hat{k}^3_{wjt} = \hat{k}^1_{wjt} - \hat{k}^2_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{55}$$

$$\hat{z}_{wjt} = 0 \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} : t < l_{wj}, \tag{56}$$

$$\hat{z}_{wjt} = 1 - \hat{k}^1_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}} : t > l_{wj}, \tag{57}$$

$$\hat{\mathcal{T}}^{\max}\hat{o}_{wj} \geq \sum_{t \in \hat{\mathcal{T}}} \hat{z}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, \tag{58}$$

$$\sum_{t \in \hat{\mathcal{T}}} t\hat{v}^1_{wjt} \leq l_{wj} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}'_w, \tag{59}$$

$$\hat{m}_{wt} \geq \hat{k}^3_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}_w, t \in \hat{\mathcal{T}}, \tag{60}$$

$$\hat{m}_{wt} \geq \hat{z}_{wjt} \quad \forall w \in \mathcal{W}, j \in \mathcal{J}^*_w, t \in \hat{\mathcal{T}}. \tag{61}$$

The behaviour of these decision variables for some job $j \in \mathcal{J}_w$ is illustrated graphically in Figure 3. In this example, maintenance begins when $t = 2$ which is after the deadline l_{wj}. This implies that the turbine will be unavailable to produce energy and consequently generating revenue will be lost until the last time that maintenance finishes at $t = 10$. Constraint sets (40)–(43) set $\hat{k}^1_{wjt}$ to 1 in all times, once maintenance on a job has started for the first time. Constraint sets (44)–(48) set $\hat{v}^1_{wjt}$ to 1 in the time when maintenance starts on a job for the first time. Constraint sets (49)–(51) set $\hat{v}^2_{wjt}$ to 1 in the time when maintenance ends on a job for the last time. Constraint sets (52)–(54) set $\hat{k}^2_{wjt}$ to 1 on all times after maintenance has ended for the last time. Constraint set (55) sets $\hat{k}^3_{wjt}$ to 1 for times between the very first start and very last finish of a job. Constraint sets (56) and (57) track the periods for each job in which the deadline is exceeded before the start of maintenance. Constraint set (58) track if a job is started after the deadline. Constraint set (59) ensures that non-failure jobs are scheduled before their deadline. Constraint sets (60)and (61) capture when the turbine is offline due to maintenance and due to a failure.

5.4. Scheduling - Objective Function

The objective is to minimise the sum of the revenue losses from performing maintenance and the costs of performing maintenance. The revenue generation of a WT in time $t \in \hat{\mathcal{T}}$ is denoted by r_t.

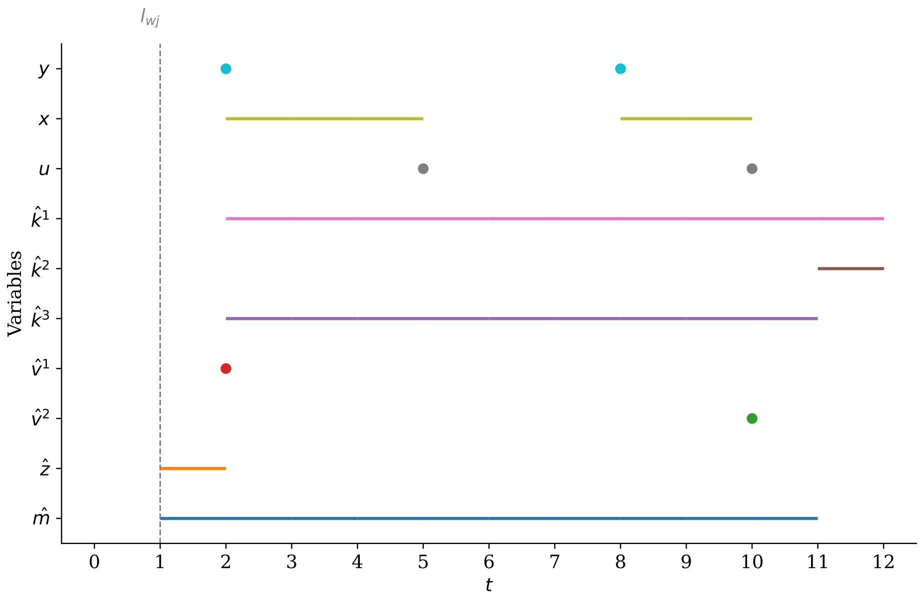

Figure 3: An example of the behaviour the decision variables for tracking turbine downtime for some job $j \in \mathcal{J}_w$.

The objective function is therefore defined as

$$\min \; \hat{S} = \sum_{w\in\mathcal{W}} \sum_{t\in\hat{\mathcal{T}}} m_{wt} r_t \tag{62}$$

$$+ \left(\sum_{w\in\mathcal{W}} \sum_{j\in\mathcal{J}_{w'}} c'_{wj} \right) + \left(\sum_{w\in\mathcal{W}} \sum_{j\in\mathcal{J}_{w^*}} \hat{o}_{wj} c^*_{wj} + \sum_{w\in\mathcal{W}} \sum_{j\in\mathcal{J}_{w^*}} (1-\hat{o}_{wj})\, c^*_{wj} \right), \tag{63}$$

where the term(62) is the total revenue loss due to turbine downtime. The maintenance costs in (63) are made up of preventative maintenance costs for all jobs that don't lead to a failure, plus the preventative costs for failure jobs that were not scheduled late, plus the corrective maintenance costs for jobs that were performed late.

5.5. Routing

As per [10, 11, 21] a set of drop-off and pick-up nodes are defined for each turbine and a leaving node and returning node for the O&M base. Let the set of drop-off nodes for the turbines be denoted by $\mathcal{W} = \{0, \ldots, n-1\}$, the set of pick-up nodes for the turbines be denoted by $\tilde{\mathcal{W}} = \{n, \ldots, 2n-1\}$. For the O&M base, the leave and return nodes are defined as $\tilde{\mathcal{W}}^{dl} = \{2n\}$ and $\tilde{\mathcal{W}}^{dr} = \{2n+1\}$, respectively. Therefore, $\tilde{\mathcal{V}} = \{0, \ldots, 2n+1\}$ is defined as the set of vertices. The set of arcs connecting the drop-off and pick-up vertices for the turbines is defined as $\tilde{\mathcal{A}}^w = \{(i,k)\ \forall i \in \mathcal{W}, k \in \{\mathcal{W} \cup \tilde{\mathcal{W}}\} : k \neq i\} \cup \{(i,k)\ \forall i \in \tilde{\mathcal{W}}, k \in \{\mathcal{W} \cup \tilde{\mathcal{W}}\} : k \neq i, k \neq i-n\}$. All routes must start at the O&M base and the set of arcs connecting the base to the drop-off vertices are defined as $\tilde{\mathcal{A}}^{dl} = \{(2n, i)\ \forall i \in \mathcal{W}\}$. All routes must also finish at the O&M base and the set of arcs connecting the pick-up vertices to the base are defined as $\tilde{\mathcal{A}}^{dr} = \{(i, 2n+1)\ \forall i \in \{\tilde{\mathcal{W}} \cup \tilde{\mathcal{W}}^{dl}\}\}$. Therefore, $\tilde{\mathcal{A}} = \tilde{\mathcal{A}}^w \cup \tilde{\mathcal{A}}^{dl} \cup \tilde{\mathcal{A}}^{dr}$, where $\tilde{\mathcal{A}}$ denotes the set of arcs that connect the drop off and pick up nodes for the turbine and the base with (i,k) as the index. The routing problem is then defined on the graph $\tilde{\mathcal{G}} = \left(\tilde{\mathcal{V}}, \tilde{\mathcal{A}}\right)$, illustrated graphically in Figure 4.

The routing is defined over the same periods as the scheduling. The travel costs are the same as for the long-term job allocation problem. The vessel also has a technician capacity limit of $\tilde{\rho}^{\max}$. The objective is to produce a set of routes that drops off and picks up the required numbers of technicians in time for their maintenance to be performed at a turbine for each shift $d \in \hat{\mathcal{D}}$ at the minimum cost. Therefore the following decision variables are introduced:

$$\tilde{x}_{ikd} = \begin{cases} 1 & \text{if arc } (i,k) \in \tilde{\mathcal{A}} \text{ is traversed in shift } d \in \hat{\mathcal{D}} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

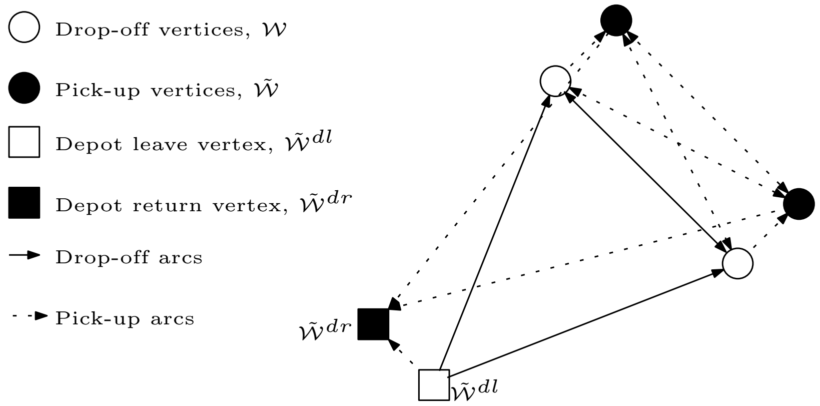

Figure 4: Example graph for a wind farm and O&M base.

$$\tilde{t}_{id} = \text{the arrival time } \tilde{t} \text{ at vertex } i \in \tilde{\mathcal{V}} \text{ on shift } d \in \hat{\mathcal{D}},$$
$$\tilde{p}_{id} = \text{the number of technicians on the vessel after leaving vertex } i \in \tilde{\mathcal{V}},$$

together with the constraints

$$\sum_{i:(i,k)\in\tilde{\mathcal{A}}^{w}\cup\tilde{\mathcal{A}}^{dl}} \tilde{x}_{ikd} = \hat{q}_{kd} \quad \forall k \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{64}$$
$$\sum_{i:(k,i)\in\tilde{\mathcal{A}}^{dl}} \tilde{x}_{kid} = \hat{q}_{kd} \quad \forall k \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{65}$$
$$\sum_{k\in\mathcal{W}\cup\tilde{\mathcal{W}}^{dr}} \tilde{x}_{ikd} = 1 \quad \forall i \in \tilde{\mathcal{W}}^{dl}, d \in \hat{\mathcal{D}}, \tag{66}$$
$$\sum_{i\in\tilde{\mathcal{W}}\cup\tilde{\mathcal{W}}^{dl}} \tilde{x}_{ikd} = 1 \quad \forall k \in \tilde{\mathcal{W}}^{dr}, d \in \hat{\mathcal{D}}, \tag{67}$$
$$\tilde{t}_{id} \geq \tilde{x}_{kid}\left(\hat{T}^{\min}_{d} + \tau_{ki}\right) \quad \forall k \in \mathcal{W}^{dl}, i \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{68}$$
$$\tilde{t}_{id} \leq \hat{T}^{\max}_{d} \quad \forall i \in \tilde{\mathcal{W}}^{dr}, d \in \hat{\mathcal{D}}, \tag{69}$$
$$\tilde{t}_{id} \geq \hat{T}^{\min}_{d} \quad \forall i \in \tilde{\mathcal{W}}^{dr}, d \in \hat{\mathcal{D}}, \tag{70}$$
$$\sum_{t\in\hat{\mathcal{T}}_d} t\hat{y}_{ijt} \geq \tilde{t}_{id} \quad \forall i \in \mathcal{W}, j \in \mathcal{J}_i, d \in \hat{\mathcal{D}}, \tag{71}$$
$$\sum_{t\in\mathcal{T}_d} (t+1)\,\hat{u}_{(i-n)jt} \leq \tilde{t}_{id} \quad \forall i \in \tilde{\mathcal{W}}, d \in \hat{\mathcal{D}}, j \in \mathcal{J}_{i-n}, \tag{72}$$
$$\tilde{t}_{id} \leq \tilde{t}_{(i+n)d} \quad \forall i \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{73}$$
$$\tilde{t}_{id} - \tilde{t}_{kd} + \tau_{ik} \leq \left|\hat{T}^{\max}_{d} - \hat{T}^{\max}_{d-1}\right|(1-\tilde{x}_{ikd}) \quad \forall (i,k) \in \mathcal{A}, d \in \hat{\mathcal{D}}, \tag{74}$$
$$\tilde{t}_{wd} \leq \hat{T}^{\max}_{d}\hat{q}_{wd} + \left|\hat{T}^{\max}_{d} - \hat{T}^{\max}_{d-1}\right|(1-\hat{q}_{wd}) \quad \forall w \in \mathcal{W}, d \in \hat{\mathcal{D}}, \tag{75}$$
$$\tilde{t}_{wd} \leq \hat{T}^{\max}_{d}\hat{q}_{(w-n)d} + \left|\hat{T}^{\max}_{d} - \hat{T}^{\max}_{d-1}\right|\left(1-\hat{q}_{(w-n)d}\right) \quad \forall w \in \tilde{\mathcal{W}}, d \in \hat{\mathcal{D}}, \tag{76}$$
$$\tilde{p}_{kd} - \tilde{p}_{id} + \hat{s}_{kd} \leq \tilde{\rho}^{\max}(1-\tilde{x}_{ikd}) \quad \forall i \in \tilde{\mathcal{W}} \cup \mathcal{W} \cup \tilde{\mathcal{W}}^{dl}, k \in \mathcal{W} \cup \tilde{\mathcal{W}}^{dr}, d \in \hat{\mathcal{D}}, \tag{77}$$
$$\tilde{p}_{kd} - \tilde{p}_{id} + \hat{s}_{kd} \geq -\tilde{\rho}^{\max}(1-\tilde{x}_{ikd}) \quad \forall i \in \tilde{\mathcal{W}} \cup \mathcal{W} \cup \tilde{\mathcal{W}}^{dl}, k \in \mathcal{W} \cup \tilde{\mathcal{W}}^{dr}, d \in \hat{\mathcal{D}}, \tag{78}$$
$$\tilde{p}_{kd} - \tilde{p}_{id} - \hat{s}_{kd} \leq \tilde{\rho}^{\max}(1-\tilde{x}_{ikd}) \quad \forall i \in \tilde{\mathcal{W}} \cup \mathcal{W} \cup \tilde{\mathcal{W}}^{dl}, k \in \tilde{\mathcal{W}}, d \in \hat{\mathcal{D}}, \tag{79}$$
$$\tilde{p}_{kd} - \tilde{p}_{id} - \hat{s}_{kd} \geq -\tilde{\rho}^{\max}(1-\tilde{x}_{ikd}) \quad \forall i \in \tilde{\mathcal{W}} \cup \mathcal{W} \cup \hat{\mathcal{W}}^{dl}, k \in \tilde{\mathcal{W}}, d \in \hat{\mathcal{D}}, \tag{80}$$
$$\tilde{p}_{id} = \tilde{p}_{(i+1)\tilde{d}} \quad \forall i \in \tilde{\mathcal{W}}^{dl}, d \in \hat{\mathcal{D}}, \tag{81}$$
$$\tilde{p}_{id} \leq \tilde{\rho}^{\max} \quad \forall i \in \tilde{\mathcal{W}}^{dl}, d \in \hat{\mathcal{D}}. \tag{82}$$

Constraint sets (64) and (65) ensure flow conservation for each node and ensure that the visits to each turbine per shift are equal to the number of scheduled visits in that shift. Constraint sets (66) and (67) stipulate that the vessel can only leave and return from the O&M base once per shift. Constraint sets (68) and (69) fix that the vessel can't leave the base before the start of a shift, and it must return to the base before the end of the shift. Constraint set (70) ensures that the vessel doesn't return to the depot before the start of a shift. Constraint sets (71)–(73) set the drop off and pick up times at each turbine, and ensure that the pick up time is after the drop off time. Constraint set (74) ensures travel time consistency with each turbine and acts as an MTZ subtour elimination constraints. Constraint sets (75) and (76) fix that the technicians must be dropped off and picked up before the end of the day. The technicians on the vessel after leaving a node are tracked using constraint sets (77)–(80). The first two constraint sets track the technicians for the drop-off nodes, whilst the latter two track the technicians for the pick-up nodes. The technicians required for the pick-up nodes are the inverse of those needed for the drop-off nodes, i.e., $\hat{s}_{id} = -\hat{s}_{(i+n)d}$. The technicians required for the O&M base nodes are set as 0. Finally constraint sets (81) and (82) ensure that the vessel returns to the base with the same number of technicians as it left with, and that there are not more technicians on the vessel than the vessel capacity limit, respectively. The objective function of the routing is then to

$$\min \ \tilde{S} = \sum_{(i,j)\in\tilde{\mathcal{A}}} \sum_{d\in\hat{\mathcal{D}}} \tilde{x}_{ijd} c_{ij}. \tag{83}$$

6. Solution Method

As the problem considered in this paper is an extension of the VRPDP it is an NP-hard problem and therefore an exact solution approach is not practical for large instances. Therefore, a matheuristic and metaheuristic, based on ALNS are proposed.

6.1. Adaptive Large Neighbourhood Search

The main steps of the solution methods are outlined in Algorithm 1. ALNS utilises the principle that large neighbourhoods contain better quality solutions than small ones and adaptively applies destroy and repair operators to a solution based on the historical performance of the operators [23]. In the proposed method, destroy and repair operators are combined into a single set of neighbourhood operators. The proposed ALNS also utilises the Metropolis acceptance criterion, commonly applied in a simulated annealing framework [22], which allows acceptance of worsening solutions so that in early iterations a large solution space is explored and in later iterations as the acceptance criteria becomes stricter and the acceptance of high-quality solutions is prioritised.

A solution is represented by, $\boldsymbol{s} = \{s^1, s^2, s^3\}$, where $s^1 = \{(w,j) : \boldsymbol{y}\}$ and $s^2 = \{(w,j) : \boldsymbol{u}\}$ are maps representing that job (w,j) $\forall w \in \mathcal{W}, j \in \mathcal{J}_w$ starts and ends at time indices $t \in T$ in the lists $\boldsymbol{y}$ and $\boldsymbol{u}$, respectively. Moreover, $s^3 = \{d : [(i, \tilde{t}_{id}), \tilde{p}_{id})]\}$ is a map containing a list of tuples representing the order in which vertices $i \in \tilde{\mathcal{V}}$ are visited together with the arrival time of the vessel at that vertex and the number of technicians on the vessel after leaving the vertex $\forall d \in \hat{\mathcal{D}}$. The objective function value denoted by $z(\boldsymbol{s})$ is an abstraction of $\hat{S} + \tilde{S} + Pe$, where Pe is the sum of penalty terms applied to infeasible solutions, described in more detail in §6.4.

An initial schedule is constructed following the steps described in Algorithm 2 (line 1) and the route for the initial schedule is constructed following the steps described in Algorithm 3 (line 2), together these form the initial solution (line 3). The initial objective function value is then computed (line 4). Then the required parameters are initialised (lines 5–11). The current best solution, and current accepted solution are denoted by $\boldsymbol{s}^*$ and $\boldsymbol{s}$, respectively. The initial temperature T_0 is set using the method described by Ben-Ameur [3] and the cooling schedule follows the geometric cooling schedule introduced by van Laarhoven & Aarts [22].The main loop of the algorithm then begins in line 12 and continues until either the maximum number of iterations, denoted by $\eta = N^{\max}$, or the number of non improving iterations denoted by η^n exceeds $N^{\max *}$. A neighbourhood operator $i \in \mathcal{I}$, where $\mathcal{I}$ denotes the set of operators, is then selected using the roulette wheel method described in Algorithm 4 (line 13). The probability that an operator $i \in \mathcal{I}$ is selected, is dependant on a minimum weighting $w_{\min}$, the total number of times that operator leads to an improved solution k_i^+, the total number of times that operator has been selected k_i, the number of iterations since the probability of acceptance has been updated η^{update} and the number of iterations after which probabilities of selection should be updated η^{reset}. In particular, η^{update} is initialised as 0. The neighbourhood operator then generates a set of neighbouring schedules denoted by $\bar{\boldsymbol{s}} \in \bar{\mathcal{S}}$ (line 15) where at this stage $\bar{\boldsymbol{s}} = \{s^1, s^2\}$. For each neighbouring schedule $\bar{\boldsymbol{s}} \in \bar{\mathcal{S}}$, the routes $\bar{s}^3$ are constructed and $\bar{\boldsymbol{s}} = \{\bar{s}^1, \bar{s}^2, \bar{s}^3\}$ is the complete neighbouring solution (lines 16-19). The neighbouring solution with the best objective value, denoted by $\bar{\boldsymbol{s}}'$, is selected (line 20), if this neighbouring solution improves the current best solution then the neighbouring solution is accepted as the new best solution (line 21). If it is worse, it is accepted as the new current solution using the Metropolis acceptance criteria (line 22). The relative iteration counters are updated and the loop continues.

Algorithm 1 Algorithmic Pseudocode for ALNS.

Input: Input data for parameters and sets, maximum number of iterations ($N_{\max}$), maximum number of non-improving iterations ($N^n_{\max}$), minimum neighbourhood weight ($w_{\min}$), iterations to update neighbourhood weights (η^{reset}), maximum nodes for shift in routing heuristic ($\tilde{n}^{\max}$), the set of neighbourhood operators ($\mathcal{I}$), matheuristic boolean

Output: The best solution $\boldsymbol{s}^*$

1: $s^1, s^2 \leftarrow$ INITIAL_SCHEDULE ▷ Algorithm 2
2: s^3, infeasibilities $\leftarrow$ ROUTING(s^1, s^2, matheuristic) ▷ Algorithm 3
3: $\boldsymbol{s} \leftarrow \{s^1, s^2, s^3\}$ ▷ Construct initial solution
4: $z^* \leftarrow z(\boldsymbol{s})$ ▷ Obtain initial objective value
5: $\boldsymbol{s}^* \leftarrow \boldsymbol{s}$
6: $T_0 \leftarrow$ *Initial Temperature*
7: $\eta^{\text{update}} \leftarrow 0$ ▷ Initialise roulette wheel parameters
8: $\eta^n \leftarrow 0$ ▷ Initialise roulette wheel parameters
9: $k_i \leftarrow 0 \ \forall i \in \mathcal{I}$ ▷ Initialise roulette wheel parameters
10: $k_i^+ \leftarrow 0 \ \forall i \in \mathcal{I}$ ▷ Initialise roulette wheel parameters
11: **while** $\eta < N_{\max}$ **and** $\eta^n < N^n_{\max}$ **do**
12: $\quad i, \eta^{\text{update}} \leftarrow$ NEIGHBOURHOOD_SELECTION($\mathcal{I}$,η,η^{update},η^{reset},$w_{\min}$,$\mathcal{K}^+$,$\mathcal{K}$, infeasibilities)
13: $\quad k_i \leftarrow k_i + 1$ ▷ Track number of times an operator is selected
14: $\quad \mathcal{S} \leftarrow$ GENERATE_NEIGHBOURS(i)
15: $\quad$ **for all** $\bar{\boldsymbol{s}} \in \mathcal{S}$ **do**
16: $\qquad \bar{s}^3$, infeasibilities $\leftarrow$ ROUTING($\bar{s}^1, \bar{s}^2$, matheuristic boolean)
17: $\qquad \bar{\boldsymbol{s}} \leftarrow \bar{\boldsymbol{s}} \cup \bar{s}^3$
18: $\quad$ **end for**
19: $\quad \boldsymbol{s}' \leftarrow \arg\min_{\bar{\boldsymbol{s}} \in \mathcal{S}} z(\bar{\boldsymbol{s}})$
20: $\quad$ **if** $z(\boldsymbol{s}') < z(\boldsymbol{s})$ **then**
21: $\qquad \boldsymbol{s} \leftarrow \boldsymbol{s}'$
22: $\qquad k_i^+ \leftarrow k_i^+ + 1$ ▷ Track successful applications of an operator
23: $\quad$ **else if** Metropolis acceptance criteria met **then** ▷ Simulated annealing
24: $\qquad \boldsymbol{s} \leftarrow \boldsymbol{s}'$
25: $\quad$ **end if**
26: $\quad$ **if** $z(\boldsymbol{s}^*) < z^*$ **then** ▷ Update best found solution
27: $\qquad \boldsymbol{s}^* \leftarrow \boldsymbol{s}$
28: $\qquad z^* \leftarrow z(\boldsymbol{s})$
29: $\qquad \eta^n \leftarrow 0$
30: $\quad$ **else**
31: $\qquad \eta^n \leftarrow \eta^n + 1$
32: $\quad$ **end if**
33: $\quad \eta \leftarrow \eta + 1$
34: **end while**
35: **return** $\boldsymbol{s}^*$

6.2. Initial Schedule

The initial solution is generated using a greedy heuristic described in Algorithm 2. First the jobs for the wind farm are sorted into a list, denoted by $\boldsymbol{j}$, by increasing deadline and then by decreasing duration such that the jobs with the earliest deadlines and longest durations are assigned first. This is denoted by the function SORTED($\mathcal{J}$) in line 3. Next a list of jobs, with unassigned start and end times, is compiled $\forall w \in \mathcal{W},\ j \in \mathcal{J}_w$, denoted by $\bar{\mathcal{J}}_w$ in line 4. A counter k is initialised as 0 in line 5

Algorithm 2 Initial Schedule.

Input: Input data for parameters and sets
Output: The initial schedule s^1, s^2

1: **procedure** INITIAL_SCHEDULE(wind farm and job parameters and sets)
2: Initialise $s^1,\ s^2$
3: $\boldsymbol{j} \leftarrow$ SORTED $(\mathcal{J})$
4: $\bar{\mathcal{J}}_w \leftarrow \{j \in \mathcal{J}_w\}\ \forall w \in \mathcal{W}$ ▷ All unassigned jobs at same turbine
5: $k = 0$
6: **while** $k < |\mathcal{J}|$ **do**
7: $j \leftarrow \boldsymbol{j}[k]$
8: $w \leftarrow w \in \mathcal{W} : j \in \mathcal{J}_w$
9: **if** $j \notin \bar{\mathcal{J}}_w$ **then** ▷ Skip assigned jobs
10: $k = k + 1$
11: **else**
12: $\mathcal{T}_a \leftarrow$ GET_ACCESSIBLE_WINDOWS(w, j)
13: **if** $\mathcal{T}_a = \emptyset$ **then** ▷ No accessible windows
14: $\mathcal{T}_a \leftarrow$ GET_ACCESSIBLE_WINDOWS_RELAXED(w, j)
15: **end if**
16: $s^1[w, j] \leftarrow [\min(\mathcal{T}_a)]$
17: $s^2[w, j] \leftarrow [\max(\mathcal{T}_a)]$
18: **if** $\left|\bar{\mathcal{J}}_w\right| > 0$ **then** ▷ Check for remaining jobs at turbine
19: **for all** $j \in \bar{\mathcal{J}}_w$ **do**
20: **if** CHECK_CONSTRAINTS$(w, j, \mathcal{T}_a)$ **then** ▷ Ensure constraints not violated
21: $s^1[w, j] \leftarrow \min(\mathcal{T}_a)$
22: $s^2[w, j] \leftarrow \min(\mathcal{T}_a) + \delta_{wj}$ ▷ Schedule for duration of job
23: $\bar{\mathcal{J}}_w \leftarrow \bar{\mathcal{J}}_w \setminus \{j\}$ ▷ Job assigned
24: **end if**
25: **end for**
26: **end if**
27: $k \leftarrow k + 1$
28: **end if**
29: **end while**
30: **return** $s^1,\ s^2$
31: **end procedure**

and the main loop of the algorithm begins in line 6 and continues until all jobs have been assigned. The job in the k-th position of $\boldsymbol{j}$ is selected and it is checked if it has already been assigned a start and end time in lines 7–9. If it has already been assigned a start time then the job is skipped (line 10). Otherwise the start time corresponding to an accessible period, which does not break technician or spare parts constraints, with minimum revenue loss and before the job deadline is found. This is denoted by the GET_ACCESSIBLE_WINDOWS function in line 12, mathematically this is expressed as

$$
\begin{aligned}
&\mathcal{T}_a = \arg\min_{t \in \hat{\mathcal{T}}} \{ \sum_{t \in \mathcal{T}_a} r_t : \\
&|\mathcal{T}_a| = \delta_{wj}, \\
&|\mathcal{T}_a| = \sum_{t \in \mathcal{T}_a} \hat{h}_{wjt}, \\
&\min t \in \mathcal{T}_a \leq l_{wj}, \\
&P_{\text{parts}}\left(\mathcal{T}_a\right) \leq \hat{\omega}^{\max}, \\
&P_{\text{techs}}\left(\mathcal{T}_a\right) \leq \hat{\rho}'_{pt} \},
\end{aligned}
$$

where $P_{parts}\left(\mathcal{T}_a\right)$ and $P_{techs}\left(\mathcal{T}_a\right)$ ensure that the number of spare parts and the number of technicians being used for all jobs assigned to that time, including the job being assigned do not exceed the capacity and availability constraints (38) and (39), respectively. If there are no feasible periods left to assign jobs that are before the job's deadline or do not break technician or spare parts constraints, i.e. $\mathcal{T}_a = \emptyset$, jobs are assigned to the accessible period with minimum revenue loss using the GET_ACCESSIBLE_WINDOWS_RELAXED function, in lines 13–15, which is the same as GET_ACCESSIBLE_WINDOWS without the deadline, spare parts or technician constraints. Once the time windows are identified, the schedule s^1 and s^2 are set for job j and turbine w in lines 16 and 17. Then if there are other jobs at the same turbine which have not been scheduled, these are scheduled to be performed at the same time if the technician and spare parts constraints are not violated, denoted as the CHECK_CONSTRAINTS function in lines 18–23, which is the same as $P_{\text{parts}}(\mathcal{T}_a)$ and $P_{\text{techs}}(\mathcal{T}_a)$. The counter is updated and the loop continues.

6.3. Routing Heuristic

For a given schedule, routes are constructed using either the exact method with relaxed time constraints implemented in CPLEX, for the matheuristic approach, or using a routing heuristic described in Algorithm 3. The routing heuristic has been developed for efficient solutions providing a route for each shift of a schedule (s^1, s^2). The algorithm begins by checking if the boolean variable matheuristic is set to true in line 2. If this is true, then a relaxed version of the routing problem, (64)–(83), is solved independently for each $d \in \hat{\mathcal{D}}$ by relaxing the constraints (69), (71), (75) and (76), in lines 3–5 using the RELAXED_EXACT_SOLVE function. In this instance the scheduling variables $\hat{q}_{wd}$, $\hat{y}_{wjt}$ and $\hat{s}_{wd}$ are all parameters determined by s^1 and s^2. If the boolean variable matheuristic is false then a simple heuristic is employed in lines 7–12. First an initial solution for the route for each day is constructed using the CONSTRUCTION_HEURISTIC function in line 8. More specifically in the construction heuristic, the drop-off nodes are first ordered by earliest required arrival time and distance from the depot. The drop-off node for the turbine with the earliest arrival time that is closest to the depot is selected as the first node on the tour. Then the technicians required for jobs at the next drop-off and all remaining drop-offs are calculated. If there are enough technicians available on the vessel to perform all remaining drop-offs the rest of the route is constructed by visiting the drop-off nodes in order and visiting the pickup nodes in reverse order. If there are not enough technicians available on the vessel to perform all remaining drop-offs the following steps are taken. If there is time to wait at the current drop-off for the maintenance to be performed before heading to the next node, the pick-up node for the current drop-off is selected as the next node in the tour. If there is not time to wait, then it is checked if there are enough technicians on vessel to visit the next drop-off node. If there are enough technicians, the next node in the tour is selected as the drop-off node for the nearest turbine. If there are not, the vessel will visit the nearest pick-up node of the turbine with the earliest finish time until there are enough technicians to perform the next drop-off. This process continues until all drop-off nodes have been assigned to the tour. The return tour is then constructed as the inverse of the drop-off tour, excluding nodes where the pick-up has already been visited. The time of arrival at each node and the number of technicians on the vessel after leaving a node are tracked.

An improvement heuristic is then used to find valid routes with a lower cost using the N-NODE_SWAP function in line 10. The 2-opt swap heuristic [1] was used but did not return satisfactory solutions. This is mostly likely due to the asymmetry of the problem. The vessel must arrive at a drop-off node before a specific time and at a pick-up node after a specific time and it must visit the pick-up node for a turbine only after it visits the drop-off node. Therefore, an n-node swap improvement heuristic is proposed, illustrated graphically in Figure 5 with only valid node combinations shown. In this method all possible combinations of n-nodes in the tour are generated, excluding the first and last nodes which must be the O&M base leave and return nodes, respectively. For each combination of n-nodes and all permutations of a tour these n-nodes are swapped and new tours are generated. Tours where the pick-up node occurs before a drop-off node are discarded and the cost of the tour including penalties is calculated. The tour with the best cost is selected. This method does not scale efficiently for very large tours, it was found for a tour with more than six drop-offs to be too computationally expensive regardless of the choice of n. Therefore, only the construction heuristic is used for tours longer than this. To avoid excessive computational time the number of nodes to be swapped is capped at $\tilde{n}^{\max}$, for this problem four was found to be an acceptable limit. Once routes are computed, if there is an infeasible route the reason for infeasibility for a given shift is returned and this reason is used to select neighbourhood operators in Algorithm 4.

6.4. Objective Function

The objective function is the same as for the MIP formulation described in §5. To explore a larger solution space infeasible solutions are permitted but are penalised in the objective function.

Algorithm 3 Routing.

Input: Input data for parameters and sets, a schedule ($\{s^1, s^2\}$), matheuristic boolean
Output: A constructed route s^3, infeasibilities

procedure ROUTING(s^1, s^2)
  if matheuristic **then**
    for all $d \in \hat{\mathcal{D}}$ **do**
      $s^3[d] \leftarrow$ RELAXED_EXACT_SOLVE(s^1, s^2)
    end for
  else
    for all $d \in \hat{\mathcal{D}}$ **do**
      $s^3[d] \leftarrow$ CONSTRUCTION_HEURISTIC(s^1, s^2)
      if Number of drop-offs on $d < 6$ **then**
        $s^3[d] \leftarrow$ N-NODE_SWAP(s^1, s^2)
      end if
    end for
  end if
  for all $d \in \hat{\mathcal{D}}$ **do**
    if Route is infeasible **then**
      Acquire reason for infeasibility
    end if
  end for
  return s^3, infeasibilities
end procedure

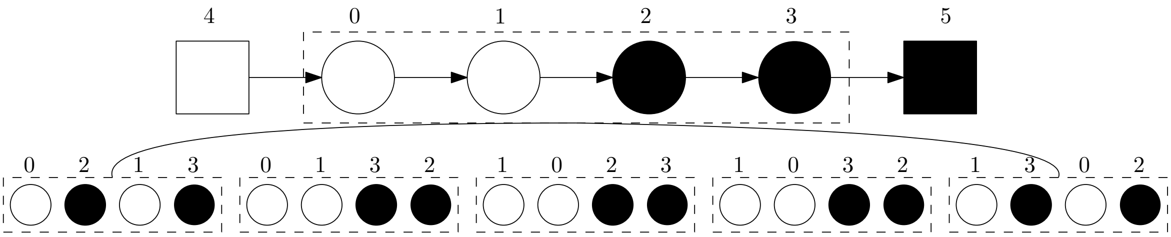

Figure 5: Example of 4-node swap heuristic for a 6 node tour with only valid combinations of swaps shown.

Jobs $j \in \mathcal{J}'_w$ which are scheduled after their deadline are penalised by Pe^{late} for each period between their deadline and start of maintenance. Each period that a job is scheduled in a time period that is inaccessible is penalised by Pe^{in}. Similarly, the technicians required; parts being used; and job location constraints are penalised by the amount they are exceeded multiplied by $Pe^{\text{tech}}, Pe^{\text{parts}}, Pe^{\text{loc}}$, respectively. If the route on any day is infeasible the reason for infeasibility is returned, in addition to penalty costs, as this guides the selection of which neighbourhood operators to apply. Infeasible routes are penalised in four ways. A late return to the O&M base is penalised by Pe^{baselate}; a late drop off at a turbine is penalised by Pe^{wtlate}; a route that is too long to perform in the shift is penalised by $Pe^{\text{routelong}}$; and insufficient technicians on the vessel to perform the next job in the tour is penalised by $Pe^{\text{routetechs}}$.

6.5. Neighbourhood Operators

Neighbouring solutions $\bar{\boldsymbol{s}} \in \mathcal{S}$ are created by employing a neighbourhood operator $i \in \mathcal{I}$. For this paper 7 different neighbourhood operators are proposed: 1-1 interchange, 1-0 interchange, slide, split, merge, split-slide and parallelisation. To explore a large neighbourhood space stacking of neighbourhoods where multiple neighbourhood operators are applied sequentially in one go is allowed. An example, based on the 1-1 interchange operator, is illustrated graphically in Figure 6.

1-1 interchange. The 1-1 interchange operator takes two jobs, in different shifts, and swaps the start times for each job, i.e., swapping the shifts on which jobs are done. Only valid swaps are considered and the jobs to be swapped are selected based on their penalties and if there are no penalties they are selected randomly. This neighbourhood is stacked with the split, merge, split-slide, parallelisation and slide operators.

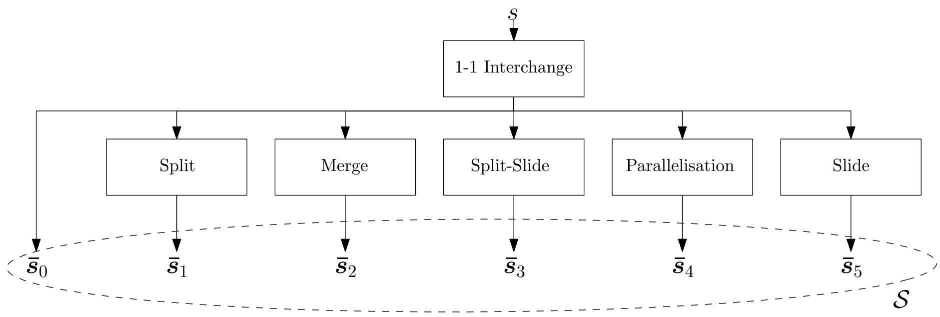

Figure 6: Example of stacking neighbourhood operators. In this example the 1-1 interchange operator generates six neighbouring schedules $\bar{s}_0$–$\bar{s}_5$. One where just the original operator is applied, $\bar{s}_0$, and five where other neighbourhood operators are subsequently applied to the neighbouring schedule $\bar{s}_0$.

1-0 interchange. The 1-0 interchange operator takes one job and moves it to another shift. The job selected is based on the one with the highest penalty costs associated and if there are no penalties it is selected randomly. The start point of the selected job is set so that the revenue loss from performing that job in the shift is minimised. This neighbourhood is stacked with the split, merge, split-slide, parallelisation and slide operators. When this neighbourhood is selected based on routing infeasibilities there is no stacking applied.

Slide. The slide operator takes one job and moves the start point of that job within its shift. The job selected is based on the one with the highest ratio between revenue losses from performing that job and the average revenue losses for a shift. The start point of the selected job is evaluated for all feasible start points within the shift. This neighbourhood is stacked with the split, merge and parallelisation operators. When this neighbourhood is selected based on routing infeasibilities there is no stacking applied and multiple jobs are shifted based on how late the arrival at each turbine is.

Split. The split operator takes a job which has more time scheduled for maintenance on a given shift than is available to perform maintenance in that shift and moves a portion of the job to the adjacent shift. This is illustrated graphically in Figure 7. The job selected is based on the one with the largest penalty for scheduling in an inaccessible period. A job can be split multiple times across multiple shifts. This neighbourhood is stacked with the 1-0 and split-slide operators.

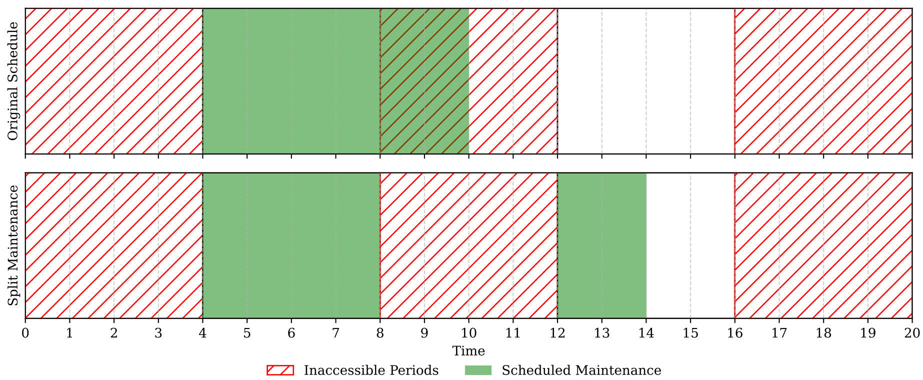

Figure 7: Example of split operator applied to a job.

Merge. The merge operator is the opposite action to the split operator. It takes a job which has been split and merges some of the splits. This neighbourhood is stacked with the 1-1, 1-0 and slide operators.

Split-Slide. The Split-Slide operator takes a job which has been split and moves the start points of each split. It does this in an identical manner to the Slide operator. This neighbourhood is stacked with the merge operator.

Parallelisation. The parallelisation operator takes a turbine with multiple jobs and schedule the jobs to occur in parallel if constraints are not violated. The principle is that performing multiple jobs on the same turbine at the same time is beneficial as the total downtime of the turbine due to maintenance for these jobs will only be equal to the duration of the longest job. This neighbourhood is stacked with the 1-0 and merge operators.

Repair. If there is an infeasible route the reason for infeasibility for a given shift is returned and a *repair* operation for that route is applied. If the reason for infeasibility is a late return to the O&M base or a late drop off at a turbine then the slide operator is chosen to be performed either on the last job before return to the O&M base or on the first late drop off, respectively. If the reason for infeasibility is a route which is too long for the accessibility window, the 1-0 interchange operator is chosen for the job with the longest travel time.

6.6. Neighbourhood Selection

Neighbourhood operators are selected according to the roulette wheel approach, where a neighbourhood operator is selected based on a probability of selection. This is described in Algorithm 4.

Algorithm 4 Neighbourhood Selection.

Input: Input data for parameters and sets, the iteration number of the ALNS (η), the number of iterations since the last update (η^{update}), the iterations after which neighbourhood weights should be updated (η^{reset}), the minimum weighting of an operator ($w_{\min}$), the number of improved solutions generated by a neighbourhood operator $i \in \mathcal{I}$ (k_i^+), the number of times that neighbourhood operator $i \in \mathcal{I}$ has been selected (k_i), infeasibilities
Output: Neighbourhood operator to use i, the number of iterations since the last update (η^{update})

1: **procedure** NEIGHBOURHOOD_SELECTION($\mathcal{I}, \eta, \eta^{\text{update}}, \eta^{\text{reset}}, w_{\min}, k_i^+, k_i$)
2:   **if** $\eta = 0$ **then**
3:     **for all** $i \in \mathcal{I}$ **do**
4:       $w_i \leftarrow \frac{1}{|\mathcal{I}|}$ ▷ Initially equal weightings
5:       Initialise p_i using (85)
6:     **end for**
7:     $i \leftarrow$ *weighted random selection*
8:   **else if** s^3 is infeasible **then**
9:     $i \leftarrow$ REPAIR(infeasibilities)
10:   **else**
11:     **if** $\eta^{\text{update}} = \eta^{\text{reset}}$ **then**
12:       **for all** $i \in \mathcal{I}$ **do**
13:         Update w_i using (84) ▷ Update weightings based on success rate
14:         Update p_i using (85)
15:       **end for**
16:       $\eta^{\text{update}} \leftarrow 0$
17:     **end if**
18:     $i \leftarrow$ *weighted random selection*
19:   **end if**
20:   $\eta^{\text{update}} \leftarrow \eta^{\text{update}} + 1$
21:   **return** i, η^{update}
22: **end procedure**

In the first iteration of the main ALNS loop all neighbourhood operators are set to have an equal probability of selection in lines 2 - 6 and a random neighbourhood operator is selected in line 7. If after the first iteration there is an infeasible route, the REPAIR operator is selected in lines 8-9. If there are no infeasible routes then, after η^{reset} iterations the probability of selection is updated based on a weighting for each neighbourhood in lines 11-17. The weighting denoted by w_i of neighbourhood operator $i \in \mathcal{I}$ is based on the success rate of each neighbourhood operator in leading to an improved solution and is determined by

$$w_i = \max\left(\frac{k_i^+}{k_i}, w_{\min}\right) \tag{84}$$

where k_i^+ is the number of improved solutions generated by a neighbourhood operator $i \in \mathcal{I}$; k_i is the number of times that neighbourhood operator $i \in \mathcal{I}$ has been selected and $w_{\min}$ is the minimum weighting for a neighbourhood operator. The probability p_i of selecting neighbourhood operator $i \in \mathcal{I}$ is then

$$p_i = \frac{w_i}{\sum_{i \in \mathcal{I}} w_i}. \tag{85}$$

A neighbourhood is then randomly selected based on these probabilities in line 18 and the number of iterations since an update η^{update} is incremented by 1 in line 20.

7. Computational Experiments

Computational experiments to evaluate the efficacy of the proposed framework and solution methods were carried out using historical data from an operational OWF in the UK. The OWF is composed of 27, 2.3 MW turbines and it is serviced using a single *crew transfer vessel* (CTV). Data were available from the maintenance logs, turbine and weather monitoring systems, historical weather forecasts and historical power prices. Two distinct experiments were undertaken:

1. A comparison of CPLEX, the ALNS metaheuristic and the ALNS matheuristic for solving the MRSP.
2. An evaluation of the full tactical and operational framework against the historical maintenance costs.

The aim of the first experiment is to evaluate the performance of the ALNS solution methods against the exact solution given by CPLEX comparing objective function value and time to reach a solution. The aim of the second experiment was to compare the solutions produced by the long-term job allocation and MRSP models within the presented framework against the historical maintenance plans to determine the effectiveness of the framework. In both experiments the following key assumptions are made:

1. Spare parts are always available.
2. Vessels are always available.
3. Technicians are always available.

Assumptions 1 and 2 were confirmed as valid by the maintenance teams at the wind farm being used for case study. Assumption 3 is made because access to the staff availability data is not possible.

The code for all experiments was written in Python with CPLEX 20.1.0.0 as the solver. The code was run on a Dell PowerEdge R740 machine running Scientific Linux 7 with an Intel Xeon Gold 6234 processor @ 3.3GHz using a maximum of 10 threads when models were solved using CPLEX and a single thread for the ALNS solution method. For both experiments hyperparameters for the ALNS method were set according to their values in Table 2. The hyperparameters were found through a grid-search hyperparameter tuning on two weeks of data. All penalty terms that are not related to routing are multiplied by the maximum revenue generation in time $t \in \hat{\mathcal{T}}$, i.e. $\arg\max_{t \in \hat{\mathcal{T}}}\{r_t\}$. Other experimental parameters can not be provided due to the confidentiality of the data.

Table 2: Tuned Hyperparameter values for the ALNS.

$N^{\max}$	2500	$N^{\max *}$	1875	$w_{\min}$	0.1
η^{reset}	125	$\tilde{n}^{\max}$	4	Pe^{late}	10
Pe^{in}	0.05	Pe^{tech}	0.05	Pe^{parts}	1.5
Pe^{loc}	1.5	Pe^{baselate}	0.1	Pe^{wtlate}	0.1
$Pe^{\text{routelong}}$	0.05	$Pe^{\text{routetechs}}$	0.1		

7.1. MRSP Solution Method Comparison

In this experiment the MRSP was solved for the 12 busiest weeks at the case study windfarm. Each week was solved independently and the maintenance jobs that were historically performed that week were the jobs input into the MRSP. The schedules and routes for each week were solved using CPLEX, ALNS metaheuristic (ALNS-m), and ALNS matheuristic (ALNS-ma). The maximum solve time for CPLEX was set to 7200s. The results of the objective function values, MIP gaps and *Lower Bound* (LB) gaps, and the run times are shown in Table 3. The objective function values are normalised against the historical baseline costs, for confidentiality reasons, so that a value of 100% means that no improvement was made over the historical maintenance schedules.

Table 3: Objective function value, MIP & LB gap, and timing results of solution methods for a 12-week period at the case-study OWF.

Week	CPLEX			ALNS-m			ALNS-ma		
	Score (%)	MIP Gap (%)	Time (s)	Score (%)	LB Gap (%)	Time (s)	Score (%)	LB Gap (%)	Time (s)
1	99.37	2.85	7202.02	98.61	2.13	62.58	98.79	2.08	735.27
2	37.92	0.01	0.92	38.95	1.28	4.15	39.15	3.24	28.82
3	22.05	0.01	100.83	22.79	2.67	12.75	22.81	3.47	5.72
4	83.61	5.65	7201.40	84.49	7.10	26.34	84.01	5.43	66.84
5	52.24	3.20	7202.70	52.06	0.79	12.21	52.20	3.22	11.08
6	56.42	4.65	7200.09	56.51	5.04	130.65	56.21	4.49	164.18
7	62.48	2.58	7201.33	60.87	0.00	71.84	60.87	0.00	7.69
8	14.65	0.00	0.87	14.65	0.00	4.52	14.65	0.00	16.80
9	16.08	0.00	12.91	16.08	0.00	5.98	16.08	0.00	35.06
10	41.03	0.00	3.82	41.03	0.00	7.09	41.60	0.62	19.92
11	63.81	1.95	7201.29	62.56	0.00	227.33	62.56	0.00	6528.94
12	97.29	2.51	7204.77	98.68	4.05	39.05	97.69	2.58	973.24
Total	50.91	-	-	50.76	-	-	50.78	-	-

It is shown that over the 12-week period all three solution methods lead to an approximately 49% improvement in total maintenance costs. In weeks 1 and 12 the historic maintenance schedule cost is very close to the costs obtained in the computational experiments. In a comparison of the solution methods the ALNS-m and ALNS-ma methods obtain equivalent or better objective function values than CPLEX for 7 out of 12-weeks but at significantly reduced solution times, in particular the ALNS-m which is 84 times faster on average. In the cases where ALNS-m and ALNS-ma outperform CPLEX it is when CPLEX was not able to reach the optimal solution within the specified time limit, indicating that these solution methods can lead to faster and higher quality solutions. In the cases where they obtain a worse objective function value the ALNS-m and ALNS-ma are on average only 0.8% absolute and 0.7% absolute worse than CPLEX, respectively. Given the solution speed and relatively quality, the ALNS-m is proposed to be used as the solution method for solving large instances of the MRSP and was selected as the solution method for the second computational experiment. The revenue losses and routing costs, for the historic maintenance schedules and the schedules produced by the three solution methods, as a proportion of total maintenance costs are presented in Table 4.

As can be seen for weeks 1 and 12 the historic revenue losses make up only 2.2% and 4.2% of the total maintenance costs, respectively, and the routing costs only make up 1.9% and 2.0% of the total maintenance costs, respectively. Given that in these tests corrective maintenance could not be performed in advance of any failures, the scope for making savings on the repair costs was limited. Therefore, for weeks when the revenue and routing costs only make up a small proportion of the overall costs, the absolute improvement of optimised maintenance schedules is small. In weeks where the revenue losses account for a larger proportion of total maintenance costs, the schedules produced by the three solution methods lead to significant savings in the total maintenance costs, for example weeks 3 and 8. Across all 12-weeks revenue losses go from accounting for 44.6% of total maintenance costs to only 6.4%, 3.8% and 4.4%. The routing costs go up from accounting for 2.8% of total maintenance costs to 4.0%, 4.4% and 4.1%; however, whilst the proportional cost of routing has increased, the absolute routing costs have decreased. The largest reduction in maintenance costs is driven by the significant reduction in revenue losses, as the routing costs make up only a small proportion of the total maintenance costs.

An example of the difference in plans between the historic and computed maintenance schedule, produced using CPLEX for week 10, are given in Figure 8. Job 2 was scheduled for the same time in both the historic and optimised plan but jobs 1 and 3 were brought forward to happen towards the

Table 4: Revenue loss and routing costs as a proportion of total maintenance costs.

Week	Revenue loss as a proportion of total costs				Routing costs as a proportion of total costs			
	CPLEX	ALNS-m	ALNS-ma	Historic	CPLEX	ALNS-m	ALNS-ma	Historic
1	1.0%	0.4%	0.3%	2.2%	1.8%	2.1%	1.9%	1.9%
2	13.2%	8.3%	7.6%	59.5%	8.1%	11.9%	12.0%	10.7%
3	7.6%	7.3%	8.8%	77.9%	6.0%	8.3%	7.5%	3.0%
4	2.1%	1.9%	1.4%	17.6%	4.6%	5.4%	4.2%	4.3%
5	4.0%	0.4%	1.5%	48.9%	4.5%	4.6%	6.4%	3.4%
6	1.7%	0.9%	0.6%	20.2%	4.8%	4.7%	4.8%	1.3%
7	4.6%	3.2%	3.0%	39.9%	3.1%	3.8%	3.8%	2.4%
8	27.4%	15.7%	20.6%	87.6%	11.1%	8.9%	12.1%	3.4%
9	20.3%	17.6%	15.6%	86.9%	10.0%	7.2%	3.9%	1.9%
10	23.9%	16.8%	21.7%	48.4%	3.5%	3.6%	2.4%	2.8%
11	5.3%	0.1%	0.1%	11.3%	3.3%	4.6%	4.4%	2.4%
12	0.2%	0.2%	0.1%	4.2%	3.3%	4.7%	3.3%	2.0%
Total	6.4%	3.8%	4.4%	44.6%	4.0%	4.4%	4.1%	2.8%

start of the week, which were lower revenue periods, rather than the end of the week. The historic weather forecasts at the start of the week revealed that the forecasted accessibility windows match those that were experienced, therefore the differences in schedules cannot be explained by inaccurate weather forecasts. A potential reason for the difference could be technician availability on a given day. Due to data unavailability this cannot be investigated.

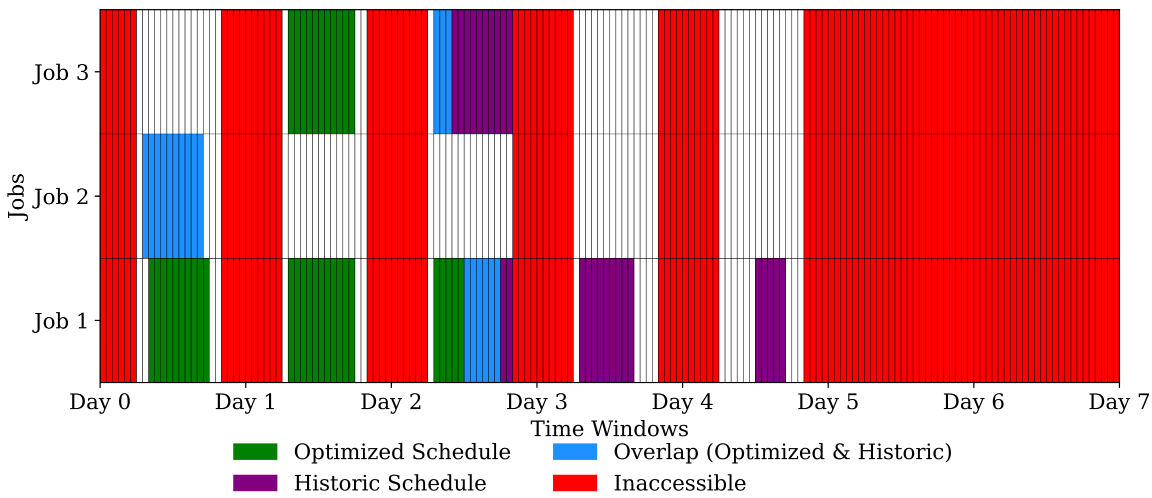

Figure 8: Example of historic and computed maintenance schedule, produced using CPLEX, for week 10.

7.2. Combined Tactical and Operational Framework Evaluation

The combined framework presented in §3 was evaluated against the historical maintenance plans for a full year of operation. This provides the longest comparison between maintenance plans, that were performed in practice, with those provided by a model. The initial long-term job allocation was first performed for the list of jobs that need to be performed at the OWF every year, for example annual services and inspections. This list was confirmed through discussion with the maintenance team at the OWF. Then the ALNS-m was used to solve the MRSP on a rolling weekly basis with new jobs appearing based on their historical notification date. Two strategies were investigated to select which solution of the resolved long-term allocation, when new jobs arose, to accept. A strategy which always accepted the minimum cost solution and one that selected the minimum disruption solution. The initial solve and all resolves of the long-term job allocation problem were solved using CPLEX. The initial solve had a 7200s time limit whilst the resolves had a 600s time limit. The revenue loss due to WT maintenance downtime in the objective function of the long-term job allocation problem is dependent on a function U_{wt} which is a function of the total utilisation of the turbine in that planning period u_{wt}. The function, an example of which is illustrated graphically in Figure 9, is non-linearly increasing with utilisation. To avoid non-linearities a piecewise linear approximation for different values of utilisation was used, an example of which is also shown in Figure 9.

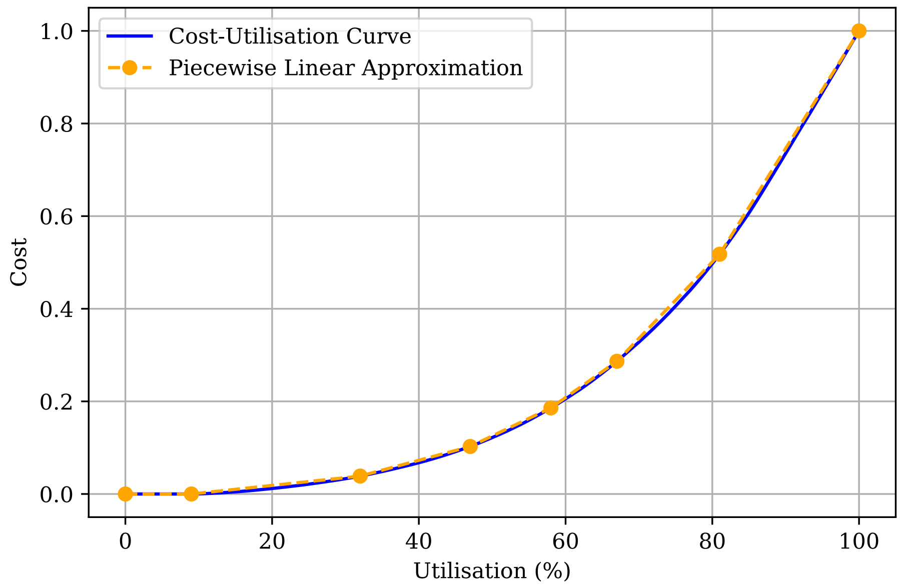

Figure 9: Example cost against utilisation curve and piecewise linear approximation.

The results of the initial long-term job allocation solution against the historical maintenance plan are shown in Figure 10. The initial long-term job allocation solve produced a plan with a cost of 81.14% of the historic plan and a MIP gap of 0.26%. As can be seen more work was prioritised towards the start of the year in the computed plan and jobs were more evenly distributed throughout the year.

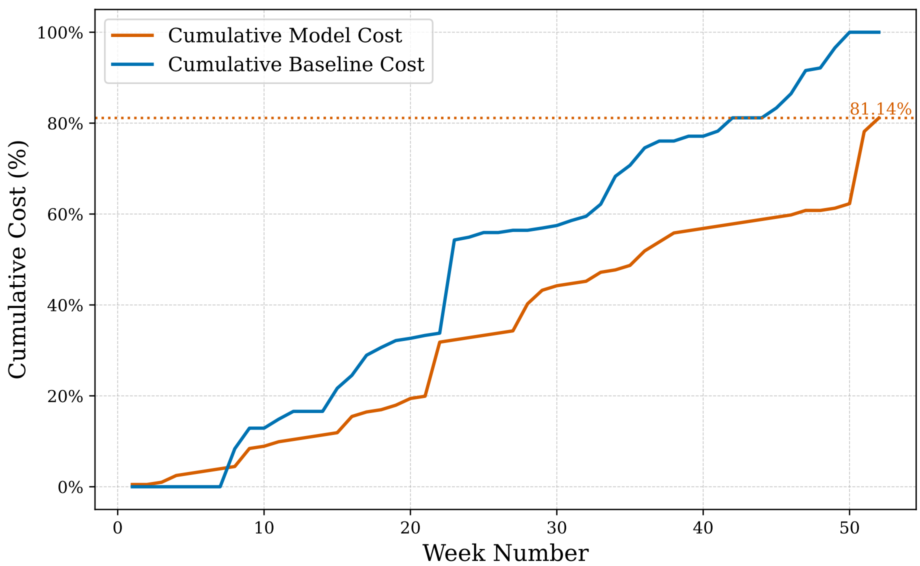

Figure 10: Chart showing the cumulative weekly cost of the computed initial long-term schedule and the historical baseline initial schedule. Only jobs considered in an initial long term plan, such as annual services, have been included.

The development of the long-term plan with new information is shown in Figure 11. Both the minimum cost strategy and minimum disruption strategy with the solution from the MRSP lead to an improvement of 66% and 65%, respectively over the historic maintenance plans. Both strategies prioritise performing work earlier in the year compared to the historical strategy. The strategy which selects the resolved long-term maintenance plans based on minimising cost leads to a larger saving compared with the strategy that selects the resolved long-term plans based on minimum disruption. This is as expected; however, the magnitude of improvement is only 1% over the course of the full year. The total cumulative disruption incurred by two strategies is shown in Figure 12. The minimum disruption strategy leads to a total cumulative disruption of approximately 20% of that of the minimum cost strategy. Given the minimal difference in costs between the two strategies, it may be preferable for the maintenance teams on site to apply a minimum disruption strategy to facilitate more consistent planning.

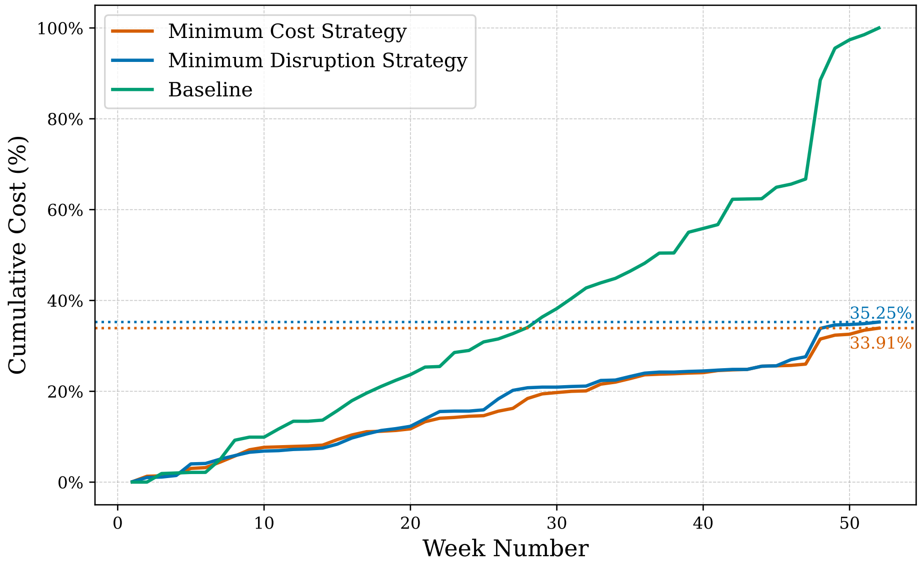

Figure 11: Chart showing the cumulative weekly cost of the computed maintenance schedules following the minimum cost and minimum disruption strategies compared with the historical baseline maintenance costs for a full year. All maintenance jobs are included.

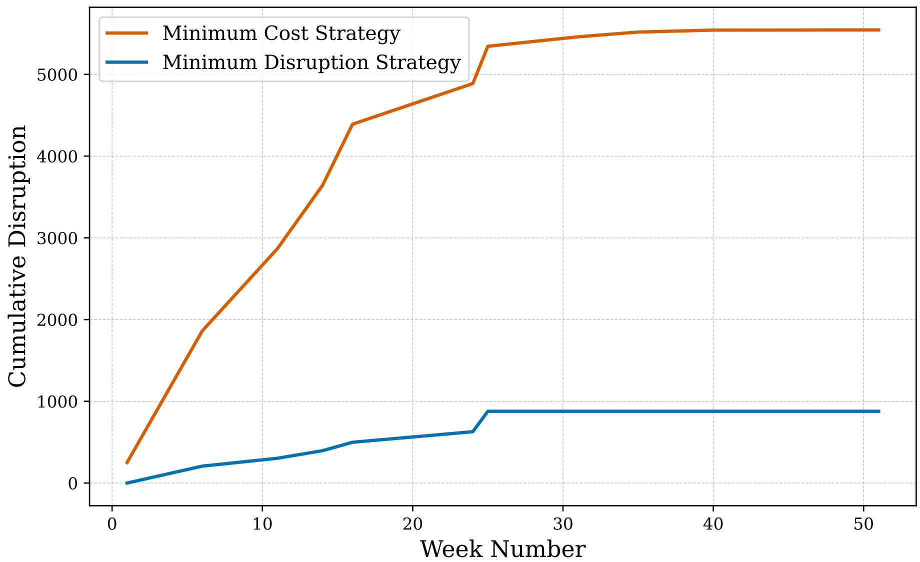

Figure 12: Chart showing the cumulative disruption of the computed maintenance schedules following the minimum cost and minimum disruption strategies.

8. Conclusions and Future Work

In this journal paper a framework, to be used by the wind farm maintenance teams, for planning OWF maintenance activities across tactical and operational timescales was presented. On the tactical timescale a long-term job allocation problem is solved to assign known maintenance jobs to periods within the year, whilst minimising the total maintenance costs including lost revenues. Then as new maintenance tasks appear and forecasts are updated, the long-term job allocation problem is re-solved, allowing the operator to choose between long-term maintenance plans that either minimise the total maintenance costs or the disruption to the original plan. On the operational timescale, based on the jobs assigned to the relevant block, a set of daily schedules and maintenance vessel routes are computed using an optimisation model where the objective function is to minimise the total cost of maintenance, including revenue losses and vessel costs. In this way the longer-term planning requirements of maintenance at offshore wind farms can be combined with the daily needs to allow more efficient and cost reducing maintenance practices. A MILP was derived and presented to solve both the long-term job allocation problem and the MRSP. A key feature of the MILP developed for the MRSP is the ability of maintenance tasks to be split across multiple maintenance shifts. An ALNS meta and matheuristic approach is also developed for solving the MRSP in a computationally efficient manner.

The results of computational experiments to determine the efficacy of the different solution methods proposed for the MRSP, reveal that the ALNS metaheuristic method is able to produce fast solutions with a lower objective function score than CPLEX for some isntances of the MRSP. In some cases where CPLEX reached the set time limit of 7200s the metaheuristic method was able to reach a solution in as little as 12s and is therefore proposed as the best method for solving the MRSP. Over a 12-week period the MRSP was able to produce maintenance schedules and routes that would have lead to a 49% reduction in maintenance costs when compared to the historic schedules. Computational experiments were also run to test the implementation of the framework for a full year of maintenance scheduling against the historic maintenance plans. The initial long-term maintenance plan was able to produce a plan with a cost 19% lower than the historical plan with a gap of 0.26% within 7200s. With the full framework combining the long-term job allocation with the MRSP using a minimum-cost and minimum-disruption strategy for selecting the resolved long-term solutions savings of 66% and 65% were achieved over the historical baselines.

The framework in this article was evaluated in deterministic conditions. In reality the planning of maintenance at OWFs is subject to much uncertainty. In particular there is, uncertainty in the weather which affects OWF accessibility windows and revenue generation; uncertainty in the electricity markets which affects the revenue generation; uncertainty in the spare parts, number of technicians and time required to complete maintenance jobs; uncertainty in the technician and spare part availability; and finally uncertainties in the occurrence of new maintenance jobs. Whilst the ability of the framework to be updated daily helps to somewhat reduce the impacts of these uncertainties an improvement to the models presented will be to directly consider the stochastic effects of the parameters mentioned in the solution stage. Finally, the next generation of OWFs will use new maintenance strategies utilising vessels such as helicopters and service operation vessels which can change the operation practices of OWF maintenance teams. Models should be developed considering the impact of these strategies on the scheduling and routing of maintenance at OWFs.

Author Statement

D Moros Conceptualisation, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Visualisation, Writing - original draft, Writing - review and editing.
K D Searle Conceptualisation, Methodology, Supervision, Writing - review and editing.
N Berrabah Conceptualisation, Supervision, Writing - review and editing.
I G Ashton Supervision, Writing - review and editing.
I Lazakis Supervision, Writing - review and editing.

Acknowledgements

The authors are grateful to EPSRC and NERC for funding for the Industrial CDT for Offshore Renewable Energy (EP/S023933/1).

Declaration of conflicting interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

Data availability

The raw and processed data resulting used for computational experiments cannot be shared at this time.

References

[1] Aarts E & Lenstra JK, 2003, *Local Search in Combinatorial Optimization*, Princeton University Press, Available from http://www.jstor.org/stable/j.ctv346t9c.

[2] Allal A, Sahnoun M, Adjoudj R, Benslimane SM & Mazar M, 2021, *Multi-agent based simulation-optimization of maintenance routing in offshore wind farms*, Computers and Industrial Engineering, **157** Manuscript 107342 (no page numbers), Available from https://doi.org/10.1016/j.cie.2021.107342.

[3] Ben-Ameur W, 2004, *Computing the initial temperature of simulated annealing*, Computational Optimization and Applications, **29**, pp. 369–385, Available from https://doi.org/10.1023/B:COAP.0000044187.23143.bd.

[4] BVG Associates, 2019, *A guide to an offshore wind farm updated and extended*, Available from http://www.thecrownestate.co.uk/guide_to_offshore_windfarm.pdf.

[5] Dai L, Stålhane M & Utne IB, 2015, *Routing and scheduling of maintenance fleet for offshore wind farms*, Wind Engineering, **39(1)**, pp. 15–30, Available from https://doi.org/10.1260/0309-524X.39.1.15.

[6] Dawid R, McMillan D & Revie M, 2016, *Development of an o&m tool for short term decision making applied to offshore wind farms*, Wind Europe Summit, (no page numbers), Available from https://strathprints.strath.ac.uk/58072/.

[7] Dawid R, McMillan D & Revie M, 2018, *Decision support tool for offshorewind farm vessel routing under uncertainty*, Energies, **11**, pp. 1–17, Available from https://doi.org/10.3390/en11092190.

[8] Fox H, Pillai AC, Friedrich D, Collu M, Dawood T & Johanning L, 2022, *A review of predictive and prescriptive offshore wind farm operation and maintenance*, Energies, **15**, pp. 1–27, Available from https://doi.org/10.3390/en15020504.

[9] Irawan CA, Eskandarpour M, Ouelhadj D & Jones D, 2021, *Simulation-based optimisation for stochastic maintenance routing in an offshore wind farm*, European Journal of Operational Research, **289**, pp. 912–926, Available from https://doi.org/10.1016/j.ejor.2019.08.032.

[10] Irawan CA, Ouelhadj D, Jones D, Stålhane M & Sperstad IB, 2017, *Optimisation of maintenance routing and scheduling for offshore wind farms*, European Journal of Operational Research, **256**, pp. 76–89, Available from https://doi.org/10.1016/J.EJOR.2016.05.059.

[11] Irawan CA, Ouelhadj D, Sperstad IB & Jones D, 2023, *A combined tactical and operational framework for maintenance scheduling and routing in offshore wind farms*, Journal of the Operational Research Society, **74**, pp. 2241–2260, Available from https://doi.org/10.1080/01605682.2022.2136543.

[12] Kuhn F, Friederike L, Matthey T, Schlosser A & Zivansky J, 2022, *How to succeed in the expanding global offshore wind market*, Available from https://www.mckinsey.com/industries/electric-power-and-natural-gas/our-insights/how-to-succeed-in-the-expanding-global-offshore-wind-market.

[13] Lazakis I & Khan S, 2021, *An optimization framework for daily route planning and scheduling of maintenance vessel activities in offshore wind farms*, Ocean Engineering, **225** Manuscript 108752 (no page numbers), Available from https://doi.org/10.1016/j.oceaneng.2021.108752.

[14] Neves-Moreira F, Veldman J & Teunter RH, 2021, *Service operation vessels for offshore wind farm maintenance: Optimal stock levels*, Renewable and Sustainable Energy Reviews, **146** Manuscript 111158 (no page numbers), Available from https://doi.org/10.1016/J.RSER.2021.111158.

[15] Raknes NT, Ødeskaug K, Stålhane M & Hvattum LM, 2017, *Scheduling of maintenance tasks and routing of a joint vessel fleet for multiple offshore wind farms*, Journal of Marine Science and Engineering, **5**, pp. 1–25, Available from `https://doi.org/10.3390/jmse5010011`.

[16] Schrotenboer A, Ursavas E & Vis I, 2019, *A branch-and-price-and-cut algorithm for resource-constrained pickup and delivery problems*, Transportation Science, **53**, pp. 1001–1022, Available from `https://doi.org/10.1287/trsc.2018.0880`.

[17] Schrotenboer AH, uit het Broek MA, Jargalsaikhan B & Roodbergen KJ, 2018, *Coordinating technician allocation and maintenance routing for offshore wind farms*, Computers & Operations Research, **98**, pp. 185–197, Available from `https://doi.org/10.1016/J.COR.2018.05.019`.

[18] Schrotenboer AH, Ursavas E & Vis IF, 2020, *Mixed integer programming models for planning maintenance at offshore wind farms under uncertainty*, Transportation Research Part C: Emerging Technologies, **112**, pp. 180–202, Available from `https://doi.org/10.1016/J.TRC.2019.12.014`.

[19] Shafiee M, 2015, *Maintenance logistics organization for offshore wind energy: Current progress and future perspectives*, Renewable Energy, **77**, pp. 182–193, Available from `https://doi.org/10.1016/j.renene.2014.11.045`.

[20] Stock-Williams C & Swamy SK, 2019, *Automated daily maintenance planning for offshore wind farms*, Renewable Energy, **133**, pp. 1393–1403, Available from `https://doi.org/10.1016/J.RENENE.2018.08.112`.

[21] Stålhane M, Hvattum LM & Skaar V, 2015, *Optimization of routing and scheduling of vessels to perform maintenance at offshore wind farms*, Energy Procedia, **80**, pp. 92–99, Available from `https://doi.org/10.1016/j.egypro.2015.11.411`.

[22] van Laarhoven PJM & Aarts EHL, 1988, *Simulated annealing : theory and applications*, 1st Edition, Kluwer Academic Publishers, Available from `https://doi.org/10.1007/978-94-015-7744-1`.

[23] Windras Mara ST, Norcahyo R, Jodiawan P, Lusiantoro L & Rifai AP, 2022, *A survey of adaptive large neighborhood search algorithms and applications*, Computers and Operations Research, **146** Manuscript 105903 (no page numbers), Available from https://doi.org/10.1016/j.cor.2022.105903.

Appendix A. Sets, Parameters and Variables

Table A.5: Sets used in this paper.

Set	Definition
Long-Term Job Assignment	
$\mathcal{T}$	The set of indices for the long-term planning periods with t as the index
$\hat{\mathcal{T}}$	The set of indices for the short-term planning periods with t as the index
$\mathcal{W}$	The set of indices for the turbines with w as the index
$\mathcal{J}_w$	The set of indices for the jobs at turbine w with j as the index
$\mathcal{J}_w^*$	The set of indices for jobs that lead to a failure of the turbine if their deadline is exceeded with j as the index
$\mathcal{J}_w'$	The set of indices for the jobs that do not lead to a failure of the turbine with j as the index
$\mathcal{R}_w$	The set indices of all job pairs that must be performed at regular intervals indexed by (j, j')
$\mathcal{F}$	The set of indices for spare parts with f as the index
$\bar{\mathcal{W}}$	The set of indices for the turbines that originally had jobs scheduled at them in the long-term resolve with w as the index
$\bar{\mathcal{J}}_w$	The set of indices for the jobs, from the original list, that have not been completed by the point of the resolve with j as the index
MRSP–Scheduling	
$\hat{\mathcal{D}}$	The set of indices for the discrete maintenance shifts with d as the index
$\hat{\mathcal{T}}_d$	The set of indices in each discrete maintenance shift with t as the index
$\hat{\mathcal{B}}$	The set of indices of the working areas in a turbine with b as the index
$\hat{\mathcal{P}}$	The set of indices of technicians types with p as the index
MRSP–Routing	
$\mathcal{W}$	The set of drop-off nodes for the turbines with w as the index
$\tilde{\mathcal{W}}$	The set of pick-up nodes for the turbines with i as the index
$\tilde{\mathcal{W}}^{dl}$	The set of leave nodes for the O&M base with i as the index
$\tilde{\mathcal{W}}^{dr}$	The set of return nodes for the O&M base with i as the index
$\tilde{\mathcal{V}}$	The set of vertices with i as the index
$\tilde{\mathcal{A}}^w$	The set of arcs that connect the drop off and pick up vertices for the turbines with (i, k) as the index
$\tilde{\mathcal{A}}^{dl}$	The set of arcs that connect the leave vertex of the base with the drop off vertices for the turbines with (i, k) as the index
$\tilde{\mathcal{A}}^{dr}$	The set of arcs that connect the pick up vertices for the turbines with the return vertex of the with (i, k) as the index
$\tilde{\mathcal{A}}$	The set of arcs that connect the drop off and pick up vertices for the turbine and the base with (i, k) as the index
ALNS	
$\mathcal{I}$	The set of neighbourhood operators with i as the index
$\bar{\mathcal{S}}$	The set of neighbouring solutions generated by a nieghbourhood operator with $\bar{\boldsymbol{s}}$ as the index

Table A.6: Parameters used in this paper.

Parameter	Definition
Long-Term Job Assignment	
$T^{\max}$	The latest long-term planning period index
$\hat{T}^{\max}$	The latest short-term planning period index
$\tau_{ww'}$	The travel time from turbine $w \in \mathcal{W}$ to turbine $w' \in \mathcal{W}$
$c_{ww'}$	The travel cost from turbine $w \in \mathcal{W}$ to turbine $w' \in \mathcal{W}$
$\bar{\tau}_w$	The travel time from the O&M base to turbine $w \in \mathcal{W}$
δ_{wj}	The time required to complete job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
l_{wj}	The deadline to complete job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
l^*_{wj}	The earliest start time for job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
$\bar{l}_{wjj'}$	The minimum interval between jobs $j \in \mathcal{J}_w$ and $j' \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
ρ_{wj}	The number of technicians required for a job $j \in \mathcal{J}_w$ on turbine $w \in \mathcal{W}$
ρ'_t	The number of technicians available in $t \in \mathcal{T}$
e_t	The amount of time available to perform maintenance in each $t \in \mathcal{T}$
α_{wjf}	The quantity of part $f \in \mathcal{F}$ required to perform a job $j \in \mathcal{J}_w$
λ_f	The lead time to acquire part $f \in \mathcal{F}$
$\bar{\alpha}_{ft}$	The initial stock of part $f \in \mathcal{F}$ in time $t \in \mathcal{T}$
c^{inv}	The inventory holding cost
c^*_{wj}	The cost to perform a corrective job $j \in \mathcal{J}^*_w$ after failure
c'_{wj}	The cost to perform a preventative job given no failure
r'_t	The cumulative revenue loss from 0 up to time $t \in \mathcal{T}$
MRSP–Scheduling	
$\hat{T}^{\min}_d$	The start index for work on each shift $d \in \hat{\mathcal{D}}$
$\hat{T}^{\max}_d$	The end time for work on each shift
$\hat{h}_{wjt}$	Binary indictor if job $j \in \mathcal{J}_w$ can be performed at turbine $w \in \mathcal{W}$ at time $t \in \hat{\mathcal{T}}$
$\hat{\beta}_{wjb}$	Binary indicator if job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$ is performed in turbine area $b \in \hat{\mathcal{B}}$
$\hat{B}^{\max}_b$	The maximum number of jobs that can be performed in any of the turbine areas at once
$\hat{\rho}_{wjp}$	The number of technicians of each type required for a job $j \in \mathcal{J}_w$ on turbine $w \in \mathcal{W}$
$\hat{\rho}'_{pt}$	The availability of technician type p in $t \in \hat{\mathcal{T}}$
$\hat{\omega}_{wj}$	The weight of spare parts required for a job $j \in \mathcal{J}_w$ on turbine $w \in \mathcal{W}$
r_t	The revenue generation of a turbine in time $t \in \hat{\mathcal{T}}$
MRSP–Routing	
$\hat{\omega}^{\max}$	The vessel spare parts capacity limit
$\tilde{\rho}^{\max}$	The vessel technician capacity limit
ALNS	
$N^{\max}$	The maximum number of ALNS iterations
$N^{\max *}$	The maximum number of non-improving ALNS iterations
$w_{\min}$	The minimum weighting of a neighbourhood operator
η^{reset}	The number of iterations after which probabilities of selection should be updated
$\boldsymbol{j}$	A sorted list of jobs
$\tilde{n}^{\max}$	The maximum number of nodes to be swapped in an n-node swap procedure
Pe^{late}	Penalty for jobs which are scheduled after their deadline
Pe^{in}	Penalty term for scheduling a job in an inaccessible period
Pe^{tech}	Penalty term for using too many technicians
Pe^{parts}	Penalty term for using too many spare parts
Pe^{loc}	Penalty term for scheduling too many jobs in a location in a turbine
Pe^{baselate}	Penalty term for a late return to the O&M base
Pe^{wtlate}	Penalty term for a late drop off at a turbine
$Pe^{\text{routelong}}$	Penalty term for a route that is too long for the shift
$Pe^{\text{routetechs}}$	Penalty term for having insufficient technicians on the vessel

Table A.7: Variables used in this paper.

Variable	Definition
Long-Term Job Assignment	
x_{wjt}	Binary variable to track if job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$ is scheduled in period $t \in \mathcal{T}$
$y_{ww't}$	Binary variable to track if turbines $w, w' \in \mathcal{W}$ are both assigned to period $t \in \mathcal{T}$
z_t	The assignment cost of period $t \in \mathcal{T}$
s_{ft}	The stock of part $f \in \mathcal{F}$ at the end of period $t \in \mathcal{T}$
r_{ft}	The quantity of part $f \in \mathcal{F}$ ordered in period $t \in \mathcal{T}$
o_{wjt}	Binary variable to track if job $j \in \mathcal{J}^*_w$ at turbine $w \in \mathcal{W}$ is delayed and assigned to period $t \in \mathcal{T}$
u_{wt}	The utilisation of turbine $w \in \mathcal{W}$ in period $t \in \mathcal{T}$
v_{wj}	The change in assignment time for $j \in \bar{\mathcal{J}}_w$ at turbine $w \in \bar{\mathcal{W}}$
MRSP–Scheduling	
$\hat{x}_{wjt}$	Binary variable to track if maintenance for job $j \in \mathcal{J}_w$ is occurring at turbine $w \in \mathcal{W}$ at time $t \in \hat{\mathcal{T}}$
$\hat{y}_{wjt}$	Binary variable to track if maintenance for job $j \in \mathcal{J}_w$ is starting at turbine $w \in \mathcal{W}$ at time $t \in \hat{\mathcal{T}}$
$\hat{u}_{wjt}$	Binary variable to track if maintenance for job $j \in \mathcal{J}_w$ ends at turbine $w \in \mathcal{W}$ at the end of time $t \in \hat{\mathcal{T}}$
$\hat{q}_{wd}$	Binary variable to track if turbine $w \in \mathcal{W}$ is visited during maintenance shift $d \in \hat{\mathcal{D}}$
$\hat{s}_{wd}$	The number of technicians needed at turbine $w \in \mathcal{W}$ for shift $d \in \hat{D}$
$\hat{k}^1_{wjt}$	Binary variable to track once job $j \in \mathcal{J}_w$ has started at turbine $w \in \mathcal{W}$ for the first time
$\hat{k}^2_{wjt}$	Binary variable to track once job $j \in \mathcal{J}_w$ has finished at turbine $w \in \mathcal{W}$ for the last time
$\hat{k}^3_{wjt}$	Binary variable to track when turbine $w \in \mathcal{W}$ is offline because of job $j \in \mathcal{J}_w$
$\hat{v}^1_{wjt}$	Binary variable to track after first start for job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
$\hat{v}^2_{wjt}$	Binary variable to track after last finish for job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$
$\hat{m}_{wt}$	Binary variable to track if turbine $w \in \mathcal{W}$ if offline in time $t \in \hat{\mathcal{T}}$
$\hat{z}_{wjt}$	Binary variable to track time $t \in \hat{\mathcal{T}}$ between a job's $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$ deadline and start if late
$\hat{o}_{wj}$	Binary variable to track if job $j \in \mathcal{J}_w$ at turbine $w \in \mathcal{W}$ is scheduled late
MRSP–Routing	
$\tilde{x}_{ikd}$	Binary variable to track if arc $(i, k) \in \tilde{\mathcal{A}}$ is traversed in shift $d \in \hat{\mathcal{D}}$
$\tilde{t}_{id}$	The arrival time $\tilde{t}$ at vertex $i \in \tilde{\mathcal{V}}$ on shift $d \in \hat{\mathcal{D}}$
$\tilde{p}_{id}$	The number of technicians on the vessel after leaving vertex $i \in \tilde{\mathcal{V}}$
ALNS	
$\boldsymbol{s}$	A solution to the ALNS
s^1	Representation of start times for jobs
s^2	Representation of end times for jobs
s^3	Representation of routing in each shift $d \in \hat{\mathcal{D}}$
η	Iteration number of ALNS
η^n	The number of non-improving iterations of ALNS
k_i^+	The number of times neighbourhood operator i leads to an improving solution
k_i	The number of times neighbourhood operator i has been selected
η^{update}	The number of iterations since the probability of acceptance has been updated
$\bar{\mathcal{J}}_w$	A list of unassigned jobs
p_i	The probability of selecting neighbourhood operator $i \in \mathcal{I}$