NEW CONTRIBUTIONS TO THE MODELLING AND OPTIMIZATION OF THE SELECTIVE MAINTENANCE PROBLEM IN COMPLEX SYSTEMS

by

Ryan Patrick O'Neil

Submitted in partial fulfillment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
August 2021

To my parents.

Table of Contents

List of Tables

List of Figures

Abstract

Selective maintenance is a strategy used in industrial and military environments where maintenance is performed between a sequence of missions. This thesis contributes to the SM literature by introducing and solving the joint selective maintenance and orienteering problem (JSMOP). The JSMOP can be applied to a wide range of systems that are geographically distributed and maintained by a crew of repair technicians. The JSMOP model will simultaneously decide the systems to visit and the maintenance actions to be performed on the visited systems to meet or exceed a target reliability.

The multimission selective maintenance problem attempts to identify the maintenance actions to be performed across multiple missions. A second contribution is made by introducing a new solution method for the multimission SMP in the form of a hybrid column generation and genetic algorithm. Through numerical experiments the hybrid column generation and genetic algorithm is shown to outperform other metaheuristics.

List of Abbreviations Used

BIP	Binary Integer Program
CM	Corrective Maintenance
CG	Column Generation
DN	Do-nothing
GA	Genetic Algorithm
GA(k, n)	k-out-of-n:G system
JSMOP	Joint Selective Maintenance and Orienteering Problem
IM	Imperfect Maintenance
MILP	Mixed Integer Linear Program
MINLP	Mixed Integer Nonlinear Program
MMSMOP	Multimission SMOP
MMSMP	Multimission SMP
MTZ	Miller-Tucker-Zemlin
MR	Minimal Repair
OP	Orienteering Problem
OWT	Offshore Wind Turbine
PM	Preventive Maintenance
RMP	Restricted Master Problem
SM	Selective Maintenance
SMOP	Selective Maintenance Optimization Problem
SMP	Selective Maintenance Problem

Acknowledgements

I would like to thank my supervisors: Dr. Claver Diallo and Dr. Abdelhakim Khatab for their continuous support and guidance. I could not imagine having better advisors and mentors for my MASc study. My sincerest gratitude is extended to Dr. Uday Venkatadri, a member of my supervisory committee, and to Dr. Samira Keivanpour, the external examiner. They have provided valuable suggestions and insightful comments. I greatly appreciate the financial support received through the Nova Scotia Energy & Mines Innovation Grant. Finally, I would like to give a special thanks to my parents and family for their continuous support and motivation.

Chapter 1

Introduction

Reliability has become a greater concern in recent years because high-tech industrial processes with ever increasing levels of sophistication are prevalent in most engineering systems (Kamal et al., 2021). To ensure that these complex systems continue to operate at expected performance levels and avoid unexpected failures, effective maintenance programs must be developed and implemented. Failures of these highly complex systems could lead to catastrophic disasters resulting in not only a loss of equipment or machinery but also in injury or loss of human life in addition to sustainability impacts. An effective maintenance program can indeed improve the reliability and/or profit of a system and simultaneously reduce potential risks and maintenance expenditures (Jiang and Liu, 2020a).

Selective maintenance (SM) is a maintenance policy used in both industrial and military environments where systems are required to perform sequences of missions separated by breaks of finite length. Examples of these systems include aerospace equipment, weapon systems, computers, production and manufacturing lines, and energy production assets. To maintain these systems and ensure they operate at an acceptable performance level during the missions, maintenance actions need to be performed during the scheduled break. Due to limited resources such as cost, time, or limited number of repair technicians it is usually not possible to perform all desirable maintenance actions during the break duration. The selective maintenance problem (SMP) seeks to answer the question of what maintenance actions to perform on what components such that either (i) the system reliability is maximized for the upcoming mission or (ii) the system meets a specified reliability for the upcoming mission at cheapest overall cost.

There are many applications where SM could be applied in an industrial setting.

Computer systems are often utilized throughout the day, but are available to be maintained after business hours, fleets of delivery vehicles are dispatched each day but are available in the evening. Production systems are required to operate 24 hours a day during weekdays but are shutdown for maintenance on weekends. Aircraft are subject to maintenance between flights with a requirement to operate at extremely high reliability levels. During the maintenance breaks there may be several systems within the asset that must be maintained to ensure that the minimum required reliability is achieved. Due to resource constraints, like time or limited technicians, not every maintenance action could be completed during the break. SM can therefore be applied to determine the maintenance actions that should be performed in order to achieve the minimum required reliability at the cheapest cost.

Due to technological developments and a desire to maintain a competitive edge, companies are expected to manage and maintain equipment and machines more effectively at lower costs. The next generation of manufacturing technology and smart factories commonly known as Industry 4.0 uses embedded systems and sensors, machine-to-machine communication, Internet of Things (IoT) and Cyber-Physical Systems (CPS) technologies to integrate the virtual space with the physical world (Xu et al., 2018; Kusiak, 2018). These new technological developments have the promise of improving the production and distribution of personalized goods at a cost similar to mass production. The supply chains, the production systems and logistic networks will be fully integrated and responsive (Kaynak, 2007; Ivanov et al., 2016). However, there will still be a need for keeping these highly interconnected systems up and running. Failures will be very costly in such networks. SM, which is often described as "doing more with less" (Cassady et al., 2001a), has the ability to provide optimal maintenance decisions that minimize the probability of failure for these highly interconnected systems under resource constraints.

The SMP has received much attention both from academic researchers and industrial practitioners since its introduction in 1998 by Rice et al. (1998). Existing SM models assume that the maintenance decisions are made and performed on systems which are all at the same location and are accessible by repair crews without resorting to any transportation means. This assumption is not always valid as industrial systems can be geographically dispersed and their maintenance may require travelling maintenance crews under transportation time and cost constraints. To fill this gap, this dissertation proposes a first contribution to the SM literature by introducing a new framework that combines the classical SMP and technician routing problem. This new framework has industrial implications as it is applicable to the maintenance of a wide range of complex systems that are geographically distributed and maintained from a pool of technicians. The second contribution of the thesis is a new heuristic solution approach based on the large-scale optimization technique of column generation combined with the genetic algorithm. Due to the highly combinatorial nature of the SMP, exact solution methods are not yet capable of solving problems of large size. This new solution method is shown to outperform other metaheuristics in terms of solution quality, and thus has the potential to provide cost savings to industries that apply this maintenance policy.

1.1 Research Objectives and Dissertation Organization

This dissertation is a thesis by articles and is comprised of two manuscripts that have been submitted for publication. Two themes are explored dealing with a novel extension and a new solution method for solving large-scale instances of the SMP. Each theme is developed in a dedicated self-contained chapter that has its own introduction, literature review, system description, model formulation, numerical experiments, and conclusion. Motivations and objectives of each research theme are summarized in the following two subsections. Lastly, the structure of the rest of the present dissertation is presented.

1.1.1 Theme 1: Optimal joint selective maintenance and orienteering

The quest for sustainable energy production is fueling the growth of offshore wind electricity generation. Energy producing offshore wind turbines are typically dispersed across several remote wind farms and must be maintained and operated with high reliability levels for long time-periods separated by scheduled maintenance rotations. Due to resource constraints such as travel time, cost, and availability of repair crews, only a subset of turbines and their components can be selected for maintenance operations during maintenance trips.

There have been several studies that have focused on the routing of maintenance technicians at offshore wind farms (Stålhane et al., 2015; Dai et al., 2015). However, the similarity between these papers is that they focus only on the routing and scheduling of technicians and rely the assumption that maintenance plan to be performed is known. López-Santana et al. (2016) investigate the combined maintenance and routing optimization problem for a set of geographically distributed machines. Jia and Zhang (2020) introduce a bi-objective optimization approach for the joint optimization of maintenance planning and workforce routing for a networked infrastructure.

Chapter 2 introduces and solves the joint selective maintenance and orienteering problem (JSMOP) with an application to offshore wind farms. Varying from other papers dealing with the combined maintenance and technician routing problem, the novel joint selective maintenance and orienteering framework proposed in this chapter focuses simultaneously on the routing, the assignment of maintenance crews, and also the detailed selection of maintenance actions, levels and components. This is a level of detail that has yet to be studied in other combined maintenance and technician routing studies. The objective of the proposed formulation is to minimize the total cost while satisfying a minimum required reliability threshold during the next operating mission until the next maintenance rotation. A mathematical optimization model is developped and fully discussed, and solution method in the form of a pattern generation algorithm is presented. Several numerical experiments demonstrate the efficiency of the solution method and the benefit of jointly optimizing selective maintenance and orienteering decisions. The SM framework presented in this chapter is applicable to a wide spectrum of complex and large systems that are geographically distributed and maintained from a pool of repair technicians.

A manuscript resulting from Theme 1 has been submitted for publication in the *European Journal of Operational Research* with submission reference EJOR-D-21-01297.

1.1.2 Theme 2: A column generation-based approach for solving the multimission selective maintenance optimization problem in large-scale serial k-out-of-n:G systems

The majority of SMP papers make the restrictive assumption that there is a single break followed by one subsequent mission. The single mission based SM plans are not optimal for systems that are required to perform multiple missions during their lifetime. The multimission SMP eliminates this restrictive assumption and attempts to identify the maintenance actions to be performed across multiple inter-mission breaks. Although the multimission SMP provides a global selective maintenance plan, it is a more complex and difficult problem to solve than the classic SMP. Because of the complexity of this optimization problem, exact solution methods cannot be used for systems of even moderate size. It is thus important to develop efficient heuristics that can find "good" solutions for moderate and large size problems.

Chapter 3 introduces a new solution method for the multimission SMP in the form of a hybrid column generation and genetic algorithm. Due to the complexity of both the system reliability and cost function, the mathematical models that have been previously suggested to solve the multimission SMP are nonlinear and usually require the use of approximate or metaheuristic-based solution methods. A solution method based on the formulation of a restricted master problem (RMP) and multiple solution generating subproblems that are solved using a genetic algorithm (GA) is presented. By integrating the GA within the classical column generation framework, high quality solutions can be obtained very quickly. The proposed solution method is capable of solving systems comprised of both parallel and k-out-of-n:G subsystems. The column generation algorithm is shown to obtain near optimal solutions and outperform other metaheuristic-based solution methods; it is also shown to be capable of solving large-scale systems comprised of many subsystems and components in a reasonable amount of time.

A manuscript resulting from Theme 2 is being reformatted for submission in *Computers and Operations Research*.

The rest of the present dissertation is structured around three additional chapters including a general conclusion chapter. The first theme is the subject of Chapter 2. This chapter investigates the joint selective maintenance and orienteering problem (JSMOP). This theme is inspired by the case of maintenance service providers tasked with servicing offshore wind turbines. A solution method in the form of a pattern generation algorithm and mathematical model is presented. Several numerical experiments are carried out to demonstrate the efficiency of the solution approach as well as its ability to achieve valid maintenance and routing decisions. Chapter 3 addresses the second theme. This chapter introduces a hybrid column generation and genetic algorithm to efficiently solve the multimission SMP for large multicomponent k-out-of$n$:G systems. The novel solution method is shown to obtain near optimal solutions. Conclusions and future research extensions are presented in chapter 4.

Chapter 2

Optimal joint selective maintenance and orienteering: A case study in offshore wind energy

2.1 Introduction

This chapter introduces and optimally solves the joint selective maintenance and orienteering problem. It is inspired by the case of a maintenance services provider tasked to service offshore wind turbines (OWTs) from a coastal city (See Figure 2.1). Wind turbines have the proven potential in supplying clean energy across the world in a sustainable and cost-effective manner. Due to the higher wind speeds, vast amounts of unoccupied space, and reduced noise pollution, offshore wind power has emerged as a promising renewable energy source and has seen rapid growth in the last decade. According to the International Energy Agency (IEA), offshore wind electricity generation increased by 32% in 2017 to reach 56 TWh and is projected to reach around 606 TWh in 2030 (IEA, 2020). One of the largest cost components of an offshore wind farm is operation and maintenance (O&M) as it accounts for approximately 25-30% of the total life cycle costs (Röckmann et al., 2017). It is of importance to develop optimal O&M plans for offshore wind farms so that the produced energy can be sold at a competitive price.

Scheduling maintenance for offshore wind farms is an extremely challenging and complicated task. OWTs are relatively more vulnerable to break downs than the onshore ones (Irawan et al., 2021), and it has been approximated that they fail 8.3 times per year (Carroll et al., 2016) on average. The resources required to maintain the turbines such as spare parts and crews of technicians are usually located at an onshore O&M base and must be transported to the designated OWTs by service vessels. Accessibility to OWTs can often be restricted due to poor weather conditions and it may not be possible to reach certain OWTs for months (Dai et al., 2015).

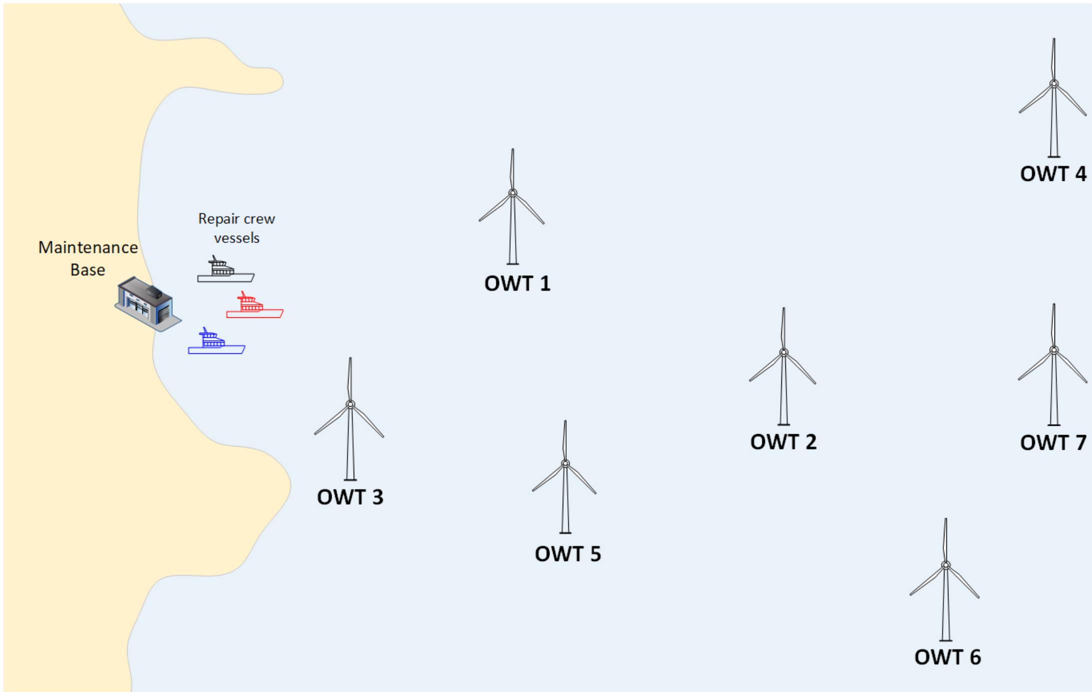

Figure 2.1: Illustration of an offshore wind farm

On average, OWTs are expected to operate at a relatively high reliability level for long periods/rotations separated by scheduled maintenance breaks. To ensure that OWTs continue to operate at the expected reliability level and to reduce random failures, maintenance actions must be performed during scheduled breaks coinciding with technician visits. During a given maintenance rotation there may be a set of turbines that have failed and also a set of turbines that have been scheduled for maintenance. Due to time and weather constraints, it may not be feasible for maintenance technicians to visit all turbines. Therefore, it is critical to identify the subset of turbines to visit, the maintenance actions that should be performed at each turbine as to meet or exceed the expected reliability level (selective maintenance problem), and also determine the optimal routes for the different crews of technicians (orienteering problem). Currently, the decision of which maintenance actions to perform and how the repair crews are routed is made sequentially: the SMP first and then the orienteering problem (OP).

The objectives of this chapter are to develop a new mathematical model to solve the joint selective maintenance and orienteering problem (JSMOP) for moderately large systems. The aim of the proposed model is to identify the components and corresponding maintenance levels to be performed on the visited locations/systems (e.g. turbines) to meet a required reliability level while also determining the optimal route for multiple repair crews. Unlike the traditional orienteering problem that aims to maximize the reward collected, the objective function of the proposed model will minimize the loss in revenue due to downtime of failed systems, travel cost between systems, maintenance cost, and a penalty cost incurred for not performing maintenance on a system (either preventive or corrective). The JSMOP is also applicable to the maintenance of a wide spectrum of complex, large, and critical systems geographically distributed and maintained from a pool of crews such as bridges, power-stations, mines, petrochemical plants, and windmills.

The remainder of this chapter is structured as follows. Section 2.2 is a review of relevant papers dealing with the SMP, OP, routing and scheduling of maintenance technician problems, and combined maintenance and routing problems. Section 2.3 lists the notation used and the main working assumptions considered. The system investigated, namely an offshore wind farm, is defined and fully described in Section 2.4. This section also presents reliability computations for an offshore wind turbine (OWT). Section 2.5 describes the maintenance modeling and total maintenance time and cost derivation for a single OWT. Section 2.6 presents the mathematical formulations of the sequential and joint optimization approaches to deal with the combination of the SM and the OP problems. Section 2.7 presents several sets of numerical experiments carried out to validate the proposed optimization models. These experiments demonstrate the benefit of jointly optimizing the SM and orienteering decisions. Conclusions and future research extensions are presented in Section 2.8.

2.2 Literature review

This review sections deals with three topics related to the objective of the proposed joint model: the SMP, the OP, and the maintenance technician problem.

2.2.1 The selective maintenance problem

Many military and industrial systems are required to perform a series of missions with finite breaks between missions. Examples of these systems include aerospace equipment, weapons, computers, production and manufacturing lines, and energy production systems. To maintain these systems and ensure they continue to operate at an acceptable performance level during the missions, maintenance actions usually need to be performed during the scheduled break intervals. The quality of maintenance action that could be performed varies from minimal repair (for failed components that are returned to the "as-bad-as-old" state) to perfect replacement (components return to the "as-good-as-new" state). Due to resource constraints such as time, cost, and repair person availability, only a subset of system components can be selected for maintenance operations. The selective maintenance problem (SMP) aims to identify the optimal subset of components and maintenance actions that will result in meeting the specified system reliability or maximizing the system reliability for the upcoming mission(s) under resource constraints.

The original selective maintenance model introduced by Rice et al. (1998) dealt with a series-parallel system with constant failure rate component and perfect repair of failed components. Due to the nonlinear objective function, an enumeration method was used to find the optimal solution. In the intervening years since Rice et al. (1998) proposed the first SMP model, many researchers have expanded upon their work. These studies have included complex system configurations (Cassady et al., 2001b; Diallo et al., 2018), multistate systems (Liu and Huang, 2010; Pandey et al., 2013a), component dependence (Xu et al., 2016; Dao and Zuo, 2017), fleet level selective maintenance (Khatab et al., 2020; Schneider and Cassady, 2015), multimission (Chaabane et al., 2020), stochastic break and/or mission duration (Liu et al., 2018; Khatab et al., 2017), condition-based SMP (Khatab et al., 2018a), and multiple repair channels (Diallo et al., 2019; Khatab et al., 2018b). A literature review of the SMP is provided in (Xu et al., 2015). A more recent SMP literature review was conducted by Cao et al. (2018).

The general goal of any SMP model is usually to maximize the reliability $\mathcal{R}$ of the system under consideration during the mission(s) following one or several intermission breaks without exceeding the total budget $\mathcal{C}_0$. A variant is to minimize the total maintenance cost $\mathcal{C}$ that guarantees a minimum required reliability level $\mathcal{R}_0$ during the subsequent mission(s). The total duration of the maintenance actions $\mathcal{T}$ should not exceed the length of the intermission break(s) $\mathcal{T}_0$. These formulations take the general forms below.

$$\begin{array}{lcl} \text{Max } \ \mathcal{R} & & \text{Min } \ \mathcal{C} \\ \text{s.t.} & & \text{s.t.} \\ \mathcal{C} \leq \mathcal{C}_0 & \qquad \text{or} \qquad & \mathcal{R} \geq \mathcal{R}_0 \\ \mathcal{T} \leq \mathcal{T}_0 & & \mathcal{T} \leq \mathcal{T}_0 \end{array}$$

The SMP has been shown by Rice (1999) to be NP-hard and therefore the variants also have this property. Hence, the large majority of papers dealing with SMP use full enumeration method for very small problems (Cassady et al., 2001a,0; Rice et al., 1998) or use nonlinear solvers or heuristics to find near-optimal solutions (Lust et al., 2009; Pandey et al., 2013b; Cao et al., 2016; Dao and Zuo, 2016; Ikonen et al., 2020). Sharma et al. (2017) combined simulation and genetic algorithm to optimize spare parts forecasting and SM decisions. Cao et al. (2017) used a simulation approach to maximize system availability for an SMP.

Diallo et al. (2018) introduced the first SMP model for serial k-out-of-n systems. The k-out-of-n system is a more complex structure than those previously considered and is a generalization of both the parallel and series configurations. A two-phase approach is developed which eliminates the need for solving a nonlinear objective function as the approach converts the SMP to a multidimensional multiple choice knapsack problem (MMKP). The first phase of the approach involves determining all possible combinations (patterns) of components and maintenance levels. The second phase consists of solving the resulting binary integer program (BIP).

All SMP papers assume that the maintenance decisions are made and performed on systems which are all at the same location and have access to ample repair crews. This is not always the case since industrial systems can be geographically distributed and their maintenance may require traveling maintenance crews under transportation time and cost constraints. Hence, the need for the proposed joint SM and orienteering problems. In the following subsection, a brief review of the OP is presented.

2.2.2 The orienteering problem

The OP can be defined on an undirected graph where the start point (vertex 1), and end point (vertex n) are fixed, and a score is given to the remaining $n-2$ vertices. The aim is to identify a path through a subset of these vertices that maximizes the total path score collected given a time constraint (Tsiligirides, 1984; Golden et al., 1987; Chao et al., 1996). The OP has been shown by Golden et al. (1987) to be NP-hard. Vansteenwegen et al. (2011) provide a mathematical model for the basic OP, as well as models for different extensions such as the team orienteering problem (TOP), and the OP with time windows. The time dependent orienteering problem (TDOP) is a variant of the original OP in which the travel time between two vertices depends on the leaving time of the first vertex. This variant was originally introduced by Fomin and Lingas (2002) and has been extended by Verbeeck et al. (2017) to include time windows. Another variant of the OP that has been studied is the capacitated TOP (CTOP) where customers have demands and vehicles have limited cargo capacity. The goal of the CTOP is to determine a path for multiple vehicles through a subset of vertices that maximizes profit collected while satisfying time limits and capacity constraints. The CTOP was introduced by Archetti et al. (2009) and extended by Archetti et al. (2013) to allow for incomplete service (CTOP-IS). Afsar and Labadie (2013) extend the classic TOP by considering that the profit collected at each customer vertex is decreasing with time.

Due to the uncertain nature of travel and service times, in addition to collected profits, some researchers have studied the OP with stochastic elements. Ilhan et al. (2008) presents the OP with scores that are normally distributed. The objective of the formulation is to maximize the probability that the profit collected in a given time limit exceeds a target profit level. Campbell et al. (2011) introduce an interesting variant of the OP in which both travel and service times are stochastic (OPSTS). The goal of the OPSTS is to identify a subset of customers and the order in which to visit them such that the expected profit is maximized. Zhang et al. (2014) present the stochastic orienteering problem with time windows where the wait time at each customer is modelled as a random variable.

The joint SM and OP problem presents unique features that differentiate it from the OP problems presented above. In particular, the time spent at a location depends on the required minimum reliability targets, the skills of the repair persons, the components selected to undergo maintenance and the maintenance levels to be carried out. Furthermore the time spent at a repair location affects the ability to visit and complete other repairs later in the trip. The sojourn time at a location is a result of optimal trade-offs between reliability, time and cost factors.

2.2.3 Combined routing and maintenance technician problems

Numerous studies have focused on the combined routing and scheduling of maintenance technicians. Tang et al. (2007) model the planned maintenance scheduling problem as a multiple tour maximum collection problem with time-dependent rewards. Kovacs et al. (2012) define the service technician routing and scheduling problem (STRSP) with and without team building and provide mathematical optimization models. The objective of their proposed models is to minimize the sum of the total routing and outsourcing costs. Goel and Meisel (2013) study the combined routing and scheduling problem for the maintenance of electricity networks. An optimization model is presented with the goal of finding an assignment of maintenance tasks to workers and a schedule for performing the tasks such that the downtime and travel costs are minimized. Mathlouthi et al. (2018) consider a multi-attribute TRSP and propose a mixed integer linear program to address the problem.

There have been a number of studies on the optimization of routing and scheduling of vessels to perform maintenance at offshore wind farms. Stålhane et al. (2015) proposed an arc-flow model with the objective of finding a route for multiple repair crews through a subset of potential offshore wind turbines that require maintenance while minimizing the overall cost of travel, maintenance, and loss in revenue due to downtime. The objective function also includes a penalty cost that is incurred when a scheduled maintenance action is not performed. It is assumed in this chapter that the fleet of service vessels is heterogeneous and the planning horizon is a 12-hour workday. Dai et al. (2015) developed a mathematical model with the aim of routing a heterogeneous fleet of service vessels to different OWTs while minimizing travel cost and a penalty cost incurred per day for delaying maintenance tasks on a turbine. This study considers a planning horizon of multiple days.

The drawback of these studies is that they focused on the routing and scheduling of maintenance technicians and vessels where the maintenance actions to be performed are assumed known and the problem is simply treated as a routing problem. López-Santana et al. (2016) present a two-step iterative approach for the combined maintenance and routing optimization problem for a set of geographically distributed machines. Jia and Zhang (2020) develop a bi-objective optimization approach for the joint optimization of maintenance planning and workforce routing for a networked infrastructure. The objective of the proposed model is to minimize the total maintenance and travel costs while satisfying reliability requirements.

Contrary to most papers dealing with the routing and scheduling of maintenance technicians, the formulation presented in this chapter focuses simultaneously on the routing, the assignment of maintenance crews, and also on the detailed selection of maintenance actions, levels and components. This is a level of detail and decision making needed in real service settings.

Before presenting the general setting of the JSMOP, the following section describes the notation used and the main working assumptions.

2.3 Notation and working assumption

2.3.1 Notation

W	Set of OWTs/locations, $W = \{1, 2, ..., N\}$
Q	Set representing the maintenance base (start and finish), $Q = \{0, N+1\}$
$\mathcal{N}$	Set of nodes defined as $\mathcal{N} = W \cup Q$, with index i

$\mathcal{A}$	Set of arcs (i, i') where $i, i' \in \mathcal{N}$
$t_{ii'}(c_{ii'})$	Time (cost) to traverse arc (i, i')
R	Set of repair crews with index r
S_i	Set of subsystems S_{ij} in OWT i, $S_i = \{1, 2, ..., N_i\}$ with index j
N_{ij}	Number of components in subsystem j of OWT i
K_{ij}	Minimum number of components that must be functioning in subsystem j of OWT i
E_{ijk}	The k^{th} component of subsystem j of OWT i
$A_{ijk}(B_{ijk})$	Effective age of component E_{ijk} after (before) maintenance
$v_{ijk}(u_{ijk})$	Operational status of component E_{ijk} after (before) maintenance
$t^c_{ijkl}(c^c_{ijkl})$	Duration (cost) of CM level l on component E_{ijk}
$t^p_{ijkl}(c^p_{ijkl})$	Duration (cost) of PM level l on component E_{ijk}
P_{ij}	Set of patterns generated for subsystem j of OWT i, $P_{ij} = \{1, \cdots, \mathcal{P}_{ij}\}$
$\mathcal{T}^{max}$	Maximum working time (max maintenance trip duration)
$\mathcal{T}_{ijp}$ ($\mathcal{C}_{ijp}$)	Time (cost) to perform maintenance plan from pattern p for subsystem j in the i^{th} OWT
c^d_i	Downtime cost rate for non-operating OWT i
t^d_i	Elapsed downtime of failed OWT i at the beginning of the maintenance trip; $t^d_i = 0$ if OWT i is working at the start of the trip
π^c_i (π^p_i)	Penalty cost per failed (functioning) OWT i that is not visited/maintained
$\mathcal{R}^c_{ijk}$	Conditional reliability of component E_{ijk}
$\mathcal{R}_{ijp}$	Reliability of subsystem j of OWT i when pattern p is selected
$\mathcal{R}_i$	Reliability of OWT i
$\mathcal{R}_0$	Required minimum OWT reliability level

2.3.2 Main working assumptions

1. Each OWT is comprised of multiple subsystems arranged in a series configuration. Each subsystem is made up of multiple binary components (meaning the components and OWT as a whole can either be functioning or failed).

2. All OWTs are identical in terms of components technologies, configuration

(RBD), and maintenance cost and time.

3. Each maintenance action requires exactly one repair crew.

4. Each service vessel carries a single repair crew. Service vessels are required to be present while maintenance is being performed and have the necessary capacity to carry any required parts/tools. There is no waiting time when maintenance is completed at a location.

5. All crews have the skills to perform any required maintenance action.

6. At the start of the maintenance trip/rotation, OWT component statuses as well as elapsed down times, and component statuses and ages are assumed known.

7. Travel times are negligible compared to component lifetimes so that component statuses do not change between maintenance crew departure and arrivals at OWT locations.

2.4 System description and reliability computation

2.4.1 System description

The system modelled is an offshore wind farm composed of N OWTs in addition to a maintenance base as illustrated in Figure 2.1. The system is viewed as a network modeled as an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N} = W \cup Q$ is the set of nodes and $\mathcal{A}$ is the set of arcs. A node $i \in \mathcal{N}$ can refer either to an OWT or the maintenance base where the repair crews must start and end their trips. Nodes numbered from 1 to N (i.e., nodes in W) refer to OWT locations, while nodes 0 and $N+1$ (i.e., nodes in Q) both refer to the same maintenance base location. An arc $(i, i^{'}) \in \mathcal{A}$ (i.e., $i, i^{'} \in \{0, \ldots, N+1\}$) is a link between nodes i and $i^{'}$, respectively, the origin and the destination. Each arc $(i, i^{'}) \in \mathcal{A}$ is characterized by the amount of time $t_{ii'}$ and cost $c_{ii'}$ incurred to traverse it.

Each OWT i is composed of N_i GA(K_{ij},N_{ij}) ($j = 1, \ldots, N_i$) subsystems in series. In reliability theory, the k-out-of-n:G configuration is usually denoted as GA(k, n)

and specifies that the system is functioning if at least k among the n components are functioning and it is a generalization of both the series and parallel structures.

At the start of a given maintenance rotation/shift, there may be a number of failed OWTs and other components from other OWTs that have been scheduled for either PM or CM. At the start of the the rotation, the exact age B_{ijk} as well as the status of every component E_{ijk} are assumed to be known. The status of component E_{ijk} before and after the maintenance visit is defined by two binary variables, respectively, u_{ijk} and v_{ijk} as follows:

$$u_{ijk} = \begin{cases} 1 & \text{if component } E_{ijk} \text{ is functioning at the start of the rotation,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

$$v_{ijk} = \begin{cases} 1 & \text{if component } E_{ijk} \text{ is functioning after the end of the rotation,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

At the start of a given rotation, the operating state of the OWT i is also defined by a binary variable U_i as:

$$U_i = \begin{cases} 1 & \text{if OWT } i \text{ has not failed at the start of the rotation ,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

2.4.2 OWT reliability computation

It is required that each OWT successfully operate the subsequent mission of duration D at a predetermined required minimum reliability level $\mathcal{R}_0$. The mission refers to the time period between two consecutive scheduled maintenance rotations. In the present work, the reliability $\mathcal{R}_i$ of an OWT i is defined by the probability of this OWT to operate during the next mission of duration D without failure. To compute the reliability of OWT i, one must first compute $\mathcal{R}^c_{ijk}$ and $\mathcal{R}_{ij}$, the reliability of component E_{ijk} and the reliability of subsystem S_{ij} respectively. Without loss of generality, it is assumed that component E_{ijk} has Weibull distributed lifetimes with the shape and scale parameters β_{ijk} and η_{ijk}, respectively.

Given that component E_{ijk} has an age A_{ijk} at the end of the maintenance rotation (i.e. at the start of the next mission), the probability that it will survive the next mission of duration D is defined by the conditional reliability function as:

$$\mathcal{R}^c_{ijk} = \exp\left[\left(\frac{A_{ijk}}{\eta_{ijk}}\right)^{\beta_{ijk}} - \left(\frac{A_{ijk}+D}{\eta_{ijk}}\right)^{\beta_{ijk}}\right]. \tag{2.4}$$

The reliability $\mathcal{R}_{ij}$ of the j^{th} subsystem of the i^{th} OWT is obtained from the exact formulation proposed in (Arulmozhi, 2002):

$$\mathcal{R}_{ij} = \sum_{e_{k_j}=1}^{N_{ij}} \sum_{e_{k_j-1}=1}^{e_{k_j}-1} \cdots \sum_{e_1=1}^{e_2-1} \left(\prod_{v=e_1}^{e_{k_j}} R^c_{ijv}\right)\left(\prod_{\substack{u=1\\ u\neq e_1,\cdots,e_{k_j}}}^{e_{k_j}} (1-R^c_{iju})\right) \tag{2.5}$$

To compute the reliability $\mathcal{R}_{ij}$ of each subsystem S_{ij}, the algorithm proposed by Kuo and Zuo (2003) and exploited in (Diallo et al., 2018) is implemented. Because the subsystems for each OWT are arranged in a serial configuration, the reliability of OWT i is computed as:

$$\mathcal{R}_i = \prod_{j\in S_i} \mathcal{R}_{ij}. \tag{2.6}$$

2.5 Maintenance modeling, and total time and cost computation

When an OWT is visited by a repair crew, the components to maintain and the maintenance actions to be performed on the selected components are known.

For each component E_{ijk}, there are $L_{ijk}+1$ maintenance levels $l \in \{0,\cdots,L_{ijk}\}$ that can be selected. These maintenance levels include do-nothing, minimal repair, imperfect maintenance (IM), and replacement. The do-nothing ($l=0$) case refers to no maintenance being performed on the component, the minimal repair ($l=1$) case if selected will return a failed component to its age just before failure. The replacement level ($l=L_{ijk}$) resets the component's age to 0, while an IM level $1<l<L_{ijk}$ if selected will return the component's age between that obtained after minimal repair

and replacement. It should be noted that only failed components are eligible to minimal repair.

Commonly used IM models in the literature are: age reduction (Malik, 1979), hazard rate adjustment (Nakagawa, 1988), and hybrid hazard rate (Lin et al., 2000) models. Without loss of generality, the age reduction approach is adopted here for IM modeling. According to this IM model, each IM level l available for component E_{ijk} is characterized by an age reduction coefficient θ_{ijkl} $(0 \leq \theta_{ijkl} \leq 1)$. Therefore, when the IM is performed on E_{ijk}, its effective age B_{ijk} is reduced and becomes:

$$A_{ijk} = \theta_{ijkl} \cdot B_{ijk}. \tag{2.7}$$

There are PM and CM durations and costs associated with each maintenance level. When performed on the component E_{ijk}, a PM of level l requires t^p_{ijkl} units of time and costs c^p_{ijkl}. Similarly, a CM of level l when carried out on the same component consumes t^c_{ijkl} unit of times and costs c^c_{ijkl} monetary units.

The total time required to maintain the i^{th} OWT and the corresponding total cost incurred are computed as:

$$\mathcal{T}_i = \sum_{j=1}^{N_i} \sum_{k=1}^{N_{ij}} \left(\sum_{l=1}^{L_{ijk}} t^c_{ijkl} \cdot (1 - u_{ijk}) \cdot \xi_{ijkl} + \sum_{l=2}^{L_{ijk}} t^p_{ijkl} \cdot u_{ijk} \cdot \xi_{ijkl} \right) \tag{2.8}$$

$$\mathcal{C}_i = \sum_{j=1}^{N_i} \sum_{k=1}^{N_{ij}} \left(\sum_{l=1}^{L_{ijk}} c^c_{ijkl} \cdot (1 - u_{ijk}) \cdot \xi_{ijkl} + \sum_{l=2}^{L_{ijk}} c^p_{ijkl} \cdot u_{ijk} \cdot \xi_{ijkl} \right) \tag{2.9}$$

where ξ_{ijkl} is a binary variable which takes the value 1 if maintenance level l is selected to be performed on component E_{ijk}, and 0 otherwise.

For each OWT that has failed, downtime costs are incurred from the instant of failure until the OWT is brought back to a functioning state through CM actions.

In addition, when an OWT undergoes preventive maintenance, a downtime cost is incurred from the instant a repair crew arrives on-site and ends when the scheduled selective maintenance plan is achieved. If the failed OWT i (i.e., $U_i = 0$ at the start of the rotation) is not visited during the maintenance rotation then a penalty cost π_i^c is incurred. Similarly, if the functioning OWT i (i.e., $U_i = 1$ at the start of the rotation) is not visited during the maintenance rotation then a penalty cost π_i^p is incurred.

2.6 Mathematical formulations

It is required that OWTs operate at a predetermined reliability level for the upcoming mission of duration D, where D refers to the time until the next scheduled maintenance rotation. Thus, the maintenance actions selected for a given OWT must result in a probability of successfully completing the next mission that is equal or greater than the required minimum reliability level $\mathcal{R}_0$. The goal of the proposed SMOP is to identify the subset of OWTs that should be visited, the corresponding components to be maintained, the levels of maintenance to be carried out, in addition to the appropriate routing that should be taken for multiple service vessels transporting the repair crews to minimize the total cost. It should be noted that, given the limited time $\mathcal{T}^{max}$ allotted to the routing maintenance, the total maintenance time $\mathcal{T}_i$, required to maintain the i^{th} OWT, must be less than or equal to an upper bound $\mathcal{T}_{0i}$ such that:

$$\mathcal{T}_{0i} = \mathcal{T}^{max} - (t_{0i} + t_{i,N+1}) \tag{2.10}$$

where t_{0i} is the required transportation time to reach OWT i from the maintenance base, while $t_{i,N+1}$ is the time required to return back.

2.6.1 Pattern generation algorithm

A maintenance pattern p is defined as a combination of components and related maintenance levels to be performed, which results in a discrete reliability value $\mathcal{R}_{ijp}$ for the j^{th} GA(K_{ij},N_{ij}) subsystem of the i^{th} OWT. The pattern generation algorithm has been demonstrated to be extremely powerful as it removes the need to solve a nonlinear objective function or nonlinear constraints (Diallo et al., 2018). The overall principle of pattern generation is given by Algorithm 1. The list of patterns obtained from the algorithm can then be used as input data for the optimization models in both sequential and joint formulations of the SMOP. The output of the algorithm is a dataset containing i, j, p, $\mathcal{R}_{ijp}$, $\mathcal{T}_{ijp}$, $\mathcal{C}_{ijp}$, and $\mathcal{P}_{ij}$ for each pattern for each subsystem in each OWT.

To illustrate how the pattern generation works, a small GA$(1, 2)$ subsystem is considered with two levels of maintenance: Do nothing ($l = 0$) and Replacement ($l = 1$). If at the start of the rotation maintenance both components are still functioning then the following list of four patterns will be generated: $(0, 0)$, $(1, 0)$, $(0, 1)$, and $(1, 1)$. Pattern $(1, 0)$ means that component 1 is replaced and no maintenance action is performed on component 2, while pattern $(1, 1)$ means that both components are replaced. If instead at the start of the shift maintenance both component are failed, only three patterns will be generated: $(1, 0)$, $(0, 1)$, and $(1, 1)$. Here, the pattern $(0, 0)$ is not eligible because both components are down and at least one of them must be replaced.

2.6.2 Sequential optimization approach

When dealing with the sequential optimization approach of the SMP and the OP, the decisions of what maintenance plan to be performed at the level of each OWT and the routing of the different repair crews are made separately. This is equivalent to first solving the SMP for all OWTs with the goal of minimizing cost subject to time constraints and reliability targets, and then solving the routing problem. The maintenance patterns that are selected for OWTs during the SMP phase are passed as input data to the routing optimization process where the decision of which OWT to visit and the path that each repair crew must take are provided. In what follows, the sequential optimization approach of the SMP and crew routing problem is presented.

Algorithm 1 Compute $\mathcal{R}_{ijp}, \mathcal{C}_{ijp}, \mathcal{T}_{ijp}$ for all valid patterns for subsystem j in the OWT i

Input data: N, N_i, K_{ij}, N_{ij}, c^c_{ijkl}, t^c_{ijkl}, c^p_{ijkl}, t^p_{ijkl}, B_{ijk}, θ_{ijkl}, β_{ijk}, η_{ijk}, $\mathcal{R}_0$, $\mathcal{T}_{0i}$
Initialize: $i = 1$
while $i \le N$ **do**
  Initialize: $j = 1$
  while $j \le N_i$ **do**
    – Generate an integer numbered set P_{ij} of all valid combination/patterns of components and their PM or CM levels such that at least K_{ij} components will be working after maintenance.
    – Find the cardinality $\mathcal{P}_{ij}$ of the set P_{ij}: $\mathcal{P}_{ij} = |P_{ij}|$.
    Initialize: $p = 1$
    while $p \le \mathcal{P}_{ij}$ **do**
      – Calculate related maintenance duration T_{ijp} by summing up all the individual durations.
      if $\mathcal{T}_{ijp} \le \mathcal{T}_{0i}$ **then**
        – Compute $\mathcal{R}^c_{ijk}$ the conditional reliability for all components E_{ijk} in the current pattern p using Equation (3.4).
        – Compute $\mathcal{R}_{ijp}$ the reliability of the subsystem j under the current pattern p using the algorithm developed in (Kuo and Zuo, 2003).
        if $\mathcal{R}_{ijp} \ge \mathcal{R}_0$ **then**
          – Calculate related maintenance cost $\mathcal{C}_{ijp}$ by summing up all the individual costs.
          – Store values of $i, j, p, \mathcal{R}_{ijp}, \mathcal{C}_{ijp}, \mathcal{T}_{ijp}$.
        end if
      else
        – Remove current pattern p from the list $\mathcal{P}_i$. (all patterns above p get shifted down by one position after the removal of p.)
        – Update $p = p - 1$ (to account for the removed pattern).
        – Update $\mathcal{P}_{ij} = |P_{ij}|$.
      end if
      $p = p + 1$.
    end while
    $j = j + 1$.
  end while
  $i = i + 1$.
end while

BIP formulation of the selective maintenance problem

The goal of the SMP is to select the maintenance plan to be performed within the time and reliability constraints such that the maintenance cost is minimized (Rice et al., 1998). To formulate the corresponding optimization problem, the following decision variable is introduced:

$$z_{ijp} = \begin{cases} 1 & \text{if pattern } p \text{ is selected for subsystem } j \text{ in OWT } i, \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

The BIP formulation for the minimization of the overall maintenance cost that must be solved for each OWT i is as follows:

$$\text{Min } \mathcal{Z}_i = \sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{C}_{ijp} \cdot z_{ijp} \tag{2.12}$$

subject to:

$$\sum_{p \in P_{ij}} z_{ijp} = 1, \quad \forall j \in S_i \tag{2.13}$$

$$\sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{T}_{ijp} \cdot z_{ijp} \leq \mathcal{T}_{0i} \tag{2.14}$$

$$\sum_{j \in S_i} \sum_{p \in P_{ij}} \ln(\mathcal{R}_{ijp}) \cdot z_{ijp} \geq \ln(\mathcal{R}_0) \tag{2.15}$$

$$z_{ijp} \in \{0, 1\}, \quad \forall i, j, p \tag{2.16}$$

In the above optimization model, constraints (2.13) ensure that a single maintenance pattern is selected for each subsystem j of OWT i. Constraints (2.14) ensure that the maintenance time at the OWT i does not exceed the alloted time window $\mathcal{T}_{0i}$ computed as the maximum working time $\mathcal{T}^{max}$ minus the sum of travel times $t_{0i} + t_{i,N+1}$ (see Eq. 2.10). Constraint (2.15) ensures that the reliability of an OWT must be greater than or equal to the target reliability. The term on the left-hand side of the inequality is the linearization of the OWT's reliability as given by Eq. (2.6):

$$\mathcal{R}_i = \prod_{j \in S_i} \mathcal{R}_{ij} \tag{2.17}$$

Because z_{ijp} is a binary variable, the equation becomes:

$$\mathcal{R}_i = \prod_{j \in S_i} \left(\sum_{p \in P_{ij}} R_{ijp} \cdot z_{ijp} \right) \tag{2.18}$$

After applying the natural logarithm which is a monotonic function to both sides of the above equation, the result in constraint (2.15) is obtained:

$$\ln(\mathcal{R}_i) = \sum_{j \in S_i} \sum_{p \in P_{ij}} \ln(\mathcal{R}_{ijp}) \cdot z_{ijp} \tag{2.19}$$

MILP formulation of the crew routing problem

Once the SMP has been first solved for each OWT i, the required maintenance time $\mathcal{T}_i$ and cost $\mathcal{C}_i$ to perform the corresponding SM plan can be obtained using Eq. (2.8) and (2.9). They can also equivalently be derived as functions of patterns as follows:

$$\mathcal{T}_i = \sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{T}_{ijp} \cdot z_{ijp}. \tag{2.20}$$

$$\mathcal{C}_i = \sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{C}_{ijp} \cdot z_{ijp}. \tag{2.21}$$

At the second step of the sequential optimization approach, the output data resulting from the SMP solutions are used to model and solve the orienteering problem. The objective of the vessel routing problem is to find a route for multiple repair vessels starting and ending at the maintenance base (nodes in $Q = \{0, N+1\}$), such that the sum of the downtime costs, travel costs, and costs associated with not performing maintenance actions within the working time is minimized. To develop the corresponding mathematical formulation, the following decision variable is introduced:

$$x_{ii'r} = \begin{cases} 1 & \text{if arc } (i, i^{'}) \text{ is traversed by repair crew } r, \\ 0 & \text{otherwise.} \end{cases} \tag{2.22}$$

In addition to the above decision variable, two other variables O_{ir} and I_{ir} are introduced. The former is used to track the arrival time of the repair crew r at the level of the i^{th} OWT, while the later (i.e., I_{ir}) serves to identify the position of node i in the travel path of the repair crew r.

The routing optimization problem is formulated as an MILP as follows:

$$
\begin{aligned}
\text{Min } \mathcal{Z} = & \sum_{i \in W} \left(\pi_i^c (1 - U_i) + \pi_i^p \cdot U_i\right) \cdot \left(1 - \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r}\right) \\
& + \sum_{i \in W} \cdot c_i^d \cdot (1 - U_i) \cdot \left(t_i^d + \sum_{r \in R} O_{ir} + \mathcal{T}_i \cdot \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r}\right) \\
& + \sum_{i \in W} c_i^d \cdot U_i \cdot \left(\mathcal{T}_i \cdot \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r}\right) + \sum_{i \in \mathcal{N} \setminus \{N+1\}} \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r} \cdot c_{ii'} \\
& + \sum_{i \in W} \mathcal{C}_i \cdot \left(\sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r}\right)
\end{aligned}
\tag{2.23}
$$

Subject to:

$$\sum_{i \in \mathcal{N} \setminus \{0\}} x_{0ir} = 1, \quad \forall r \in R \tag{2.24}$$

$$\sum_{i \in \mathcal{N} \setminus \{N+1\}} x_{i,N+1,r} = 1, \quad \forall r \in R \tag{2.25}$$

$$\sum_{i \in \mathcal{N} \setminus \{N+1\}} x_{ii'r} = \sum_{i'' \in \mathcal{N} \setminus \{0\}} x_{i'i''r}, \quad \forall r \in R; \forall i' \in W \tag{2.26}$$

$$\sum_{i \in \mathcal{N} \setminus \{N+1\}} \sum_{r \in R} x_{ii'r} \leq 1, \quad \forall i' \in W \tag{2.27}$$

$$O_{ir} \geq 0, \quad \forall i \in \mathcal{N}; \forall r \in R \tag{2.28}$$

$$O_{i'r} \leq \mathcal{T}^{max} \cdot \sum_{i \in \mathcal{N} \setminus \{N+1\}} x_{ii'r}, \quad \forall i' \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.29}$$

$$O_{0r} = 0, \quad \forall r \in R \tag{2.30}$$

$$O_{0r} + t_{0i'} - O_{i'r} - M_{0i'} \cdot (1 - x_{0i'r}) \leq 0, \quad \forall i' \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.31}$$

$$O_{ir} + t_{ii'} + \mathcal{T}_i - O_{i'r} - M_{ii'} \cdot (1 - x_{ii'r}) \leq 0, \quad \forall i \in W; \forall i' \in \mathcal{N}; \forall r \in R \tag{2.32}$$

$$2 \leq I_{ir} \leq |\mathcal{N}|, \quad \forall i \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.33}$$

$$I_{ir} - I_{i'r} + 1 \leq (|\mathcal{N}| - 1) \cdot (1 - x_{ii'r}), \quad \forall i \in W; \forall i' \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.34}$$

$$x_{ii'r} \in \{0, 1\}, \quad \forall i, i', r \tag{2.35}$$

The objective function (2.23) minimizes the total cost resulting from repair crews maintenance activities and transportation over the distributed wind turbine farm. The first term represents the penalty cost incurred when maintenance is not performed on failed or functioning OWTs. The second term represents the increasing penalty as time passes for failed systems. The third term represents the downtime cost associated with performing PM. The fourth term represents the transportation cost, while the fifth term represents the cost of performing the selected maintenance actions. Constraints (2.24) and (2.25) ensure that the route for each service vessel starts and ends at the maintenance base. Constraints (2.26) ensure the connectivity of the route, and constraints (2.27) ensure that each vertex (OWT) is visited at most once by all repair crews. Constraints (2.28) – (2.32) track the arrival time at each vertex of the distributed network $\mathcal{G}$. Inequalities (2.33) and (2.34) are the Miller-Tucker-Zemlin subtour elimination constraints where $M_{ii'}$ are real numbers of high values. Constraints (2.35) are binary variables restrictions.

2.6.3 Joint selective maintenance and orienteering problem (JSMOP)

Unlike the sequential model discussed above, the proposed joint selective maintenance and orienteering problem (JSMOP) allows simultaneous decision-making of the optimal routes for multiple repair crews together with the optimal selective maintenance plan to be performed at the level of each visited OWT. In contrast to the classic orienteering problem that aims to maximize the reward collected at each vertex (node), the objective function of the proposed JSMOP formulation will be to minimize the total cost resulting from maintenance activities, repair crews duties and transportation. In this chapter, fixed and variable labor costs of repair crews are account for in the maintenance cost. The resulting JSMOP provides joint optimal solutions to three optimization problem, namely the selective maintenance problem, the repair crews assignment problem, and the vessels routing problem. Let c_r^f and c_r^v represent the fixed and variable repair crew cost. To distinguish between repair crews' qualification levels, repair crew r is characterized by its speed ϵ_r ($\epsilon_r > 0$) to perform a maintenance action. For example, $\epsilon_r = 1$ means that the repair crew r carries out the maintenance action in the standard/baseline time duration, while $\epsilon_r = 0.5$ means that the repair crew is capable of performing the same maintenance action in half the standard duration. It is assumed that faster repair crews have higher costs.

To formulate the JSMOP, the variables O_{ir} and I_{ir} introduced previously are used to track the arrival time of the repair crew r at each OWT i and identify the position of node i in the travel path of the repair crew r respectively. The decision variable in Eq. (2.22) will also be used. Two additional decision variables z_{ijpr} and y_r are also introduced and defined as follows:

$$z_{ijpr} = \begin{cases} 1 & \text{if pattern } p \text{ is selected for subsystem } j \text{ in OWT } i, \\ & \text{ and performed by repair crew } r, \\ 0 & \text{otherwise.} \end{cases} \tag{2.36}$$

$$y_r = \begin{cases} 1 & \text{if repair crew } r \text{ is used,} \\ 0 & \text{otherwise} \end{cases} \tag{2.37}$$

The mathematical JSMOP is formulated as follows:

$$\begin{aligned} \text{Min } \mathcal{Z} = & \sum_{i \in W} \left(\pi_i^c (1 - U_i) + \pi_i^p \cdot U_i\right) \cdot \left(1 - \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r}\right) \\ & + \sum_{i \in W} c_i^d \cdot (1 - U_i) \cdot \left(t_i^d + \sum_{r \in R} O_{ir} + \sum_{j \in S_i} \sum_{p \in P_{ij}} \sum_{r \in R} \mathcal{T}_{ijp} \cdot z_{ijpr} \cdot \epsilon_r\right) \\ & + \sum_{i \in W} c_i^d \cdot U_i \cdot \left(\sum_{j \in S_i} \sum_{p \in P_{ij}} \sum_{r \in R} z_{ijpr} \cdot \mathcal{T}_{ijp} \cdot \epsilon_r\right) \\ & + \sum_{i \in W} \sum_{j \in S_i} \sum_{p \in P_{ij}} \sum_{r \in R} \mathcal{C}_{ijp} \cdot z_{ijpr} \\ & + \sum_{i \in \mathcal{N} \setminus \{N+1\}} \sum_{i' \in \mathcal{N} \setminus \{0\}} \sum_{r \in R} x_{ii'r} \cdot c_{ii'} + \sum_{r \in R} \left(O_{N+1,r} \cdot c_r^v + y_r \cdot c_r^f\right) \end{aligned} \tag{2.38}$$

s.t.:

$$\sum_{i \in \mathcal{N} \setminus \{0\}} x_{0ir} = 1, \quad \forall r \in R \tag{2.39}$$

$$\sum_{i \in \mathcal{N} \setminus \{w+1\}} x_{i,N+1,r} = 1, \quad \forall r \in R \tag{2.40}$$

$$\sum_{i \in \mathcal{N} \setminus \{N+1\}} x_{ii'r} = \sum_{i'' \in \mathcal{N} \setminus \{0\}} x_{i'i''r}, \quad \forall r \in R; \forall i^{'} \in W \tag{2.41}$$

$$\sum_{i \in \mathcal{N} \setminus \{N+1\}} \sum_{r \in R} x_{ii'r} \leq 1, \quad \forall i^{'} \in W \tag{2.42}$$

$$y_r \geq x_{ii'r}, \quad \forall i \in \mathcal{N} \setminus \{N+1\}; \forall i^{'} \in W; \forall r \in R \tag{2.43}$$

$$O_{ir} \geq 0, \quad \forall i \in \mathcal{N}; \forall r \in R \tag{2.44}$$

$$O_{i'r} \leq \mathcal{T}^{max} \cdot \sum_{i \in \mathcal{N} \setminus \{N+1\}} x_{ii'r}, \quad \forall i^{'} \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.45}$$

$$O_{0r} = 0, \quad \forall r \in R \tag{2.46}$$

$$O_{0r} + t_{0i'} - O_{i'r} - M_{0i'} \cdot (1 - x_{0i'r}) \leq 0, \quad \forall i^{'} \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.47}$$

$$\begin{aligned} O_{ir} + t_{ii'} &+ \sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{T}_{ijp} \cdot z_{ijpr} \cdot \epsilon_r \\ &- O_{i'r} - M_{ii'} \cdot (1 - x_{ii'r}) \leq 0, \quad \forall i \in W; \forall i^{'} \in \mathcal{N} \setminus \{0\}; \forall r \in R \end{aligned} \tag{2.48}$$

$$2 \leq I_{i'r} \leq |\mathcal{N}|, \quad \forall i^{'} \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.49}$$

$$I_{ir} - I_{i'r} + 1 \leq (|\mathcal{N}| - 1) \cdot (1 - x_{ii'r}), \quad \forall i \in W; \forall i^{'} \in \mathcal{N} \setminus \{0\}; \forall r \in R \tag{2.50}$$

$$\sum_{p \in P_{ij}} z_{ijpr} = \sum_{i' \in \mathcal{N} \setminus \{0\}} x_{ii'r}, \quad \forall i \in W; \forall j \in S_i; \forall r \in R \tag{2.51}$$

$$\sum_{j \in S_i} \sum_{p \in P_{ij}} \sum_{r \in R} \ln(\mathcal{R}_{ijp}) \cdot z_{ijpr} \geq \ln(\mathcal{R}_0), \quad \forall i \in W \tag{2.52}$$

$$x_{ii'r} \in \{0,1\}, \quad \forall i, i^{'}, r \tag{2.53}$$

$$y_r \in \{0,1\}, \quad \forall r \tag{2.54}$$

$$z_{ijpr} \in \{0,1\}, \quad \forall i, j, p, r \tag{2.55}$$

The objective function (2.38) minimizes the total cost of performing maintenance on the distributed systems. The first term represents the penalty cost incurred when maintenance is not performed on failed or functioning turbines. The second term represents the increasing penalty for failed turbines.The third term is the downtime cost associated with performing PM. The fourth term computes the cost of performing the selective maintenance plan, while the fifth term is the transportation cost. The last term of the objective function refers to the summation of the fixed cost and variable labor cost for hiring and using the repair crews. Constraints (2.39) and (2.40) ensure that the route for each service vessel starts and ends at the maintenance base. Constraints (2.41) ensure the connectivity of the route, and constraint (2.42) ensures that each vertex is visited at most once. Constraints (2.43) ensure that repair crews are available before they can be assigned to maintenance routes. Constraints (2.44) – (2.48) track the arrival time at each vertex and constraints (2.49) and (2.50) prevent subtours. $M_{ii'}$ are large positive numbers (i.e. big M values). Constraints (2.51) ensure that a single maintenance pattern is selected for every subsystem of each OWT, however no maintenance pattern will be selected for OWTs that are not visited. Constraints (2.52) ensure that, after maintenance actions are performed, the reliability of visited OWTs must be greater than or equal to the required minimum reliability level. Constraints (2.53) – (2.55) define the binary decision variables $x_{ii'r}$, $y_r$and z_{ijpr} used in the formulation.

2.7 Numerical experiments

In this section, five sets of numerical experiments are conducted using the case and data from a maintenance firm providing services to an offshore wind farm. The first set is a validation experiment to ensure the proposed extended model obtains the same results as previously published articles. The second experiment demonstrates the benefit of selecting an appropriately scaled value for the big M values used in the formulation in constraints (2.47) and (2.48). The third experiment compares the proposed joint model and the sequential model. The fourth and fifth experiments demonstrate additional features and advantages of the JSMOP optimization model.

All experiments are run on a Intel™i5 2.9GHz desktop computer with 12GB of RAM running Windows 10™. All algorithms were coded in Python 3.8. The optimization runs were carried out by Gurobi 9.1 using gurobipy.

2.7.1 Case study

All experiments, except the validation experiment (#1), are carried out using an offshore wind farm composed of $N = 7$ identical OWTs as illustrated in Figure 2.1. As depicted in Figure 2.2, the RBD of turbine i ($i = 1, \cdots, 7$) is composed of $S_i = 4$ k-out-of-n:G subsystems GA$(2, 5)$, GA$(2, 4)$, GA$(1, 3)$ and GA$(1, 2)$ in series. The shape and scale parameters, and operational states of each OWT components are given in the data table in Appendix A. This table lists also the cost and duration for each maintenance level. To carry out maintenance activities, there are $R = 3$ identical repair crews with fixed and variable costs set to \$150 and \$25 respectively. The required travel times and costs between OWT locations and the maintenance base are given in Table 2.1 and table 2.2. The other parameters are: $\pi_i^c = \$12,500$, $\pi_i^p = \$10,000$, and $c_i^d = \$162$/hour. According to data in the table in Appendix A, OWTs in locations 3, 5 and 7 are failed at the start of maintenance rotation. The values of elapsed down times t_i^d are provided in Table 2.3. The duration of the next mission (i.e., the period that the OWTs operate until the next maintenance rotation) is of six months (i.e., $D = 6$ months).

Table 2.1: Traveling time (hours) between turbines and maintenance base

	End vertex							
Start vertex	0	1	2	3	4	5	6	7
0	0	0.47	0.61	0.54	0.69	0.59	0.76	0.71
1	0.47	0	0.15	0.12	0.25	0.12	0.27	0.25
2	0.61	0.15	0	0.20	0.17	0.10	0.15	0.12
3	0.54	0.12	0.20	0	0.34	0.10	0.29	0.32
4	0.69	0.25	0.17	0.34	0	0.27	0.25	0.12
5	0.59	0.12	0.10	0.10	0.27	0	0.20	0.22
6	0.76	0.27	0.15	0.29	0.25	0.20	0	0.12
7	0.71	0.25	0.12	0.32	0.12	0.22	0.12	0

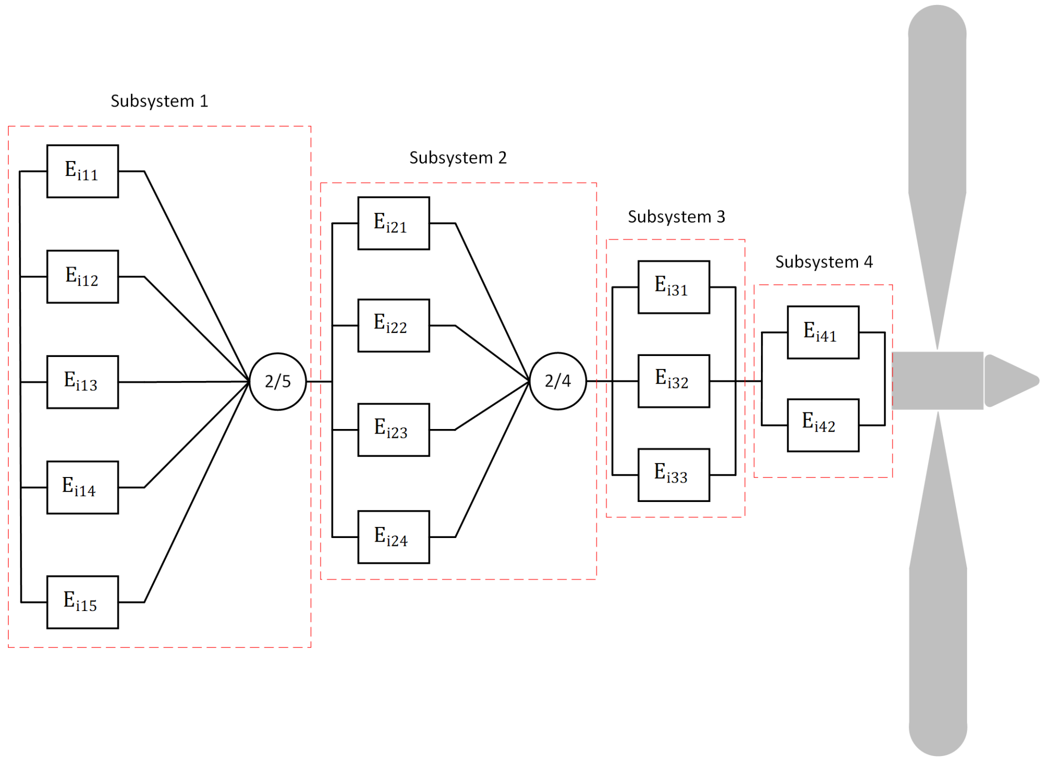

Figure 2.2: Reliability structure of the OWT at location i

Table 2.2: Traveling cost ($) between turbines and maintenance base

Start vertex	End vertex							
	0	1	2	3	4	5	6	7
0	0	17.60	22.84	20.22	25.84	22.10	28.46	26.59
1	17.60	0	5.62	4.49	9.36	4.49	10.11	9.36
2	22.84	5.62	0	7.49	6.37	3.75	5.62	4.49
3	20.22	4.49	7.49	0	12.73	3.75	10.86	11.98
4	25.84	9.36	6.37	12.73	0	10.11	9.36	4.49
5	22.10	4.49	3.75	3.75	10.11	0	7.49	8.24
6	28.46	10.11	10.86	10.86	9.36	7.49	0	4.49
7	26.59	9.36	11.98	11.98	4.49	8.24	4.49	0

Table 2.3: Elapsed downtime t_i^d of failed OWTs

OWT	t_i^d(hours)
3	48
5	10
7	2

For all numerical experiments four potential maintenance levels for failed components will be considered: do-nothing ($l = 0$), minimal repair ($l = 1$), imperfect maintenance ($l = 2$) and replacement ($l = 3$). The same maintenance levels will be considered for functioning components with the exception of minimal repair. It will be assumed that the imperfect maintenance level will reduce the age of the component by half ($\theta_{ijk2} = 0.5$). Except for the last set of experiments, the repair crews are assumed to be of standard qualification with $\epsilon_r = 1,\ \forall r \in R$.

2.7.2 Set of experiments #1: validation of the JSMOP model

This set of experiments considers the complex system comprised of three GA(k, n) subsystems GA$(2, 5)$, GA$(3, 8)$ and GA$(4, 10)$ from experiment set #5 in Diallo et al. (2018). The goal is to show that the proposed JSMOP model finds the same optimal maintenance decisions as in Diallo et al. (2018). Given that the JSMOP is a more general model than the one in Diallo et al. (2018), several parameters should be set to zero to allow for a valid comparison to take place. In particular, we set $i = 1$ because there is only one equipment in the reference experiment and all routing times and costs are set to 0. The proposed JSMOP is solved for a required minimum reliability level $\mathcal{R}_0 = 0.70$, and different values of $\mathcal{T}^{max}$. The overall results obtained are reported in Table (2.4) along with the optimal results from Diallo et al. (2018). From Table (2.4), one can indeed conclude that the proposed JSMOP model reaches the same solutions as in Diallo et al. (2018).

Table 2.4: Results of validation experiment with k-out-of-n:G subsystems

$\mathcal{T}^{max}$	Results from JSMOP			From Diallo et al. (2018)		
	$\mathcal{Z}^*$ (\$)	$\mathcal{T}^*$ (hr)	$\mathcal{R}^*$ (%)	$\mathcal{Z}^*$ (\$)	$\mathcal{T}^*$ (hr)	$\mathcal{R}^*$ (%)
100	147	64	70.84	147	64	70.84
60	153	59	71.56	153	59	71.56
56	154	56	70.18	154	56	70.18

2.7.3 Set of experiments #2: selecting the value of parameter M in JSMOP

For computational efficiency, M should be as small as possible while ensuring that constraints (2.47) and (2.48) are enforced. From constraint (2.47), given that $O_{0r} = 0$, one may conclude that $t_{0i'}$ is the smallest possible value that $M_{0i'}$ could take. Since O_{ir} can take values within the time interval $[0, \mathcal{T}^{max}]$, it follows that values of $M_{ii'}$ in constraints (2.48) must satisfy the following inequalities:

$$M_{ii'} \geq \mathcal{T}^{max} + t_{ii'} + \sum_{j \in S_i} \sum_{p \in P_{ij}} \mathcal{T}_{ijp} \cdot z_{ijpr} \cdot \epsilon_r, \quad \forall r \in R.$$

From the above inequalities, the smallest values of $M_{ii'}$ are thus obtained as:

$$M_{ii'} = \mathcal{T}^{max} + t_{ii'} + \max_{\{r \in R\}} \epsilon_r \cdot \sum_{j \in S_i} \max_{\{p \in P_{ij}\}} \mathcal{T}_{ijp}. \tag{2.56}$$

Several experiments were conducted where the JSMOP is solved using both a large value ($M = 100,000$) and the values suggested by Equation (2.56). The results of the different experiments with varying target reliabilities and shift lengths are shown in Table 2.5. The results clearly show that the computation time (CPUt) reduces significantly for all problem instances when the values suggested by Equation (2.56) are used.

Table 2.5: CPUt comparison

		Values of M from Eq.(2.56)		$M = 100,000$	
$\mathcal{R}_0(\%)$	$\mathcal{T}^{max}$	$\mathcal{Z}^*$ (\$)	CPUt (s)	$\mathcal{Z}^*$ (\$)	CPUt (s)
96.0	15	19,656	**100.27**	19,656	132.98
98.0	15	41,341	**311.48**	41,341	383.84
98.0	18	31,818	**370.12**	31,818	501.92
99.5	25	64,692	**260.97**	64,692	392.22

2.7.4 Set of experiments #3: comparison between the JSMOP formulation and the sequential model

In this set of experiments, the sequential and joint SMOP optimization approaches are compared. The same OWT farm and its corresponding input data as described in experiment #2 are considered. The optimization problems resulting from both the sequential and joint formulations are solved for varying required minimum reliability levels and $\mathcal{T}^{max} = 19$ hours. The fixed and variable costs for the repair crews are set to \$0 as repair crews are not considered for the current experiments. The results obtained are reported in Table (2.6) which gives, for each value of the target reliability, the optimal cost $\mathcal{Z}^*$ and the total number N^* of the OWTs visited by the repair crews. From these results, it is clear that the proposed joint optimization approach achieves equal or in most cases better solutions than the sequential optimization approach. Indeed, the joint approach allows not only to select the best combination of maintenance patterns to meet the target reliability at the lowest cost, but also to select more expensive patterns that either allow visiting more OWTs or lead to a reduction in OWT maintenance time which in turn may reduce the overall incurred cost.

Table 2.6: Comparison of optimal values

	Joint approach (JSMOP)		Sequential approach		Cost reduction
$\mathcal{R}_0$(%)	$\mathcal{Z}^*$ (\$)	N^*	$\mathcal{Z}^*$ (\$)	N^*	by JSMOP (%)
98.0	**21,589**	**7**	30,207	6	28.53
98.8	**44,916**	**5**	51,316	4	12.47
99.0	51,962	4	51,962	4	–
99.2	**60,607**	3	60,809	3	0.33

As expected, the results in Table (2.6) show that as the required minimum reliability $\mathcal{R}_0$ increases, the total repair time and cost increase. As a result, the total number N^* of OWTs visited per maintenance rotation decrease while the total cost incurred increase. For all $\mathcal{R}_0$ values, the joint approach achieves a lower or equal objective function value (cost) with at least as many or more OWT visits.

Tables (2.7) and (2.8) give the maintenance patterns selected and the resulting total maintenance cost $\mathcal{C}_i$ and time $\mathcal{T}_i$ for each turbine i respectively for $\mathcal{R}_0 = 98.80\%$.

For example, the first row of Table (2.7) shows the maintenance pattern selected for the first turbine: components E_{112}, E_{114}, and E_{122} are selected for minimal repair (maintenance level $l = 1$), components E_{124} and E_{132} are selected for imperfect maintenance (maintenance level $l = 2$), and the component E_{141} is selected for a replacement (maintenance level $l = 3$). The remaining components are not maintained. These maintenance actions take a total time of $\mathcal{T}_1 = 9.25$ hours and cost $\mathcal{C}_1 = \$590$. In Table 2.7, turbines 3 and 7 have a corresponding total maintenance time and a cost equal to zero meaning that they are not visited during the maintenance rotation. For the sequential approach, Table 2.8 shows the maintenance patterns selected for all OWTs followed by the decisions of what OWTs to visit and the routes that the repair crews should take.

Table 2.7: Patterns and resulting total time and cost: case of the joint optimization approach

OWT i	Patterns selected	Cost $\mathcal{C}_i$ ($)	Time $\mathcal{T}_i$ (hr)
1	(0, 1, 0, 1, 0), (0, 1, 0, 2), (0, 2, 0), (3, 0)	590	9.25
2	(0, 1, 0, 0, 0), (0, 2, 1, 2), (0, 0, 0), (3, 0)	550	9.25
3	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	0	0
4	(0, 2, 1, 0, 0), (0, 1, 0, 1), (0, 0, 0), (2, 0)	400	7.00
5	(0, 1, 0, 0, 0), (0, 2, 0, 1), (2, 0, 0), (3, 1)	640	11.0
6	(0, 1, 0, 0, 1), (0, 0, 0, 2), (0, 2, 0), (3, 0)	515	8.25
7	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	0	0

Table 2.8: Patterns and resulting time and cost: case of the sequential optimization approach

OWT i	Patterns selected	Cost $\mathcal{C}_i$ ($)	Time $\mathcal{T}_i$ (hr)	Visited
1	(0, 1, 0, 1, 0), (0, 2, 0, 0), (0, 2, 0), (3, 0)	565	9.75	
2	(0, 2, 0, 0, 0), (0, 0, 2, 2), (0, 0, 0), (3, 0)	525	9.50	
3	(0, 3, 0, 0, 1), (0, 3, 0, 0), (0, 2, 0), (1, 3)	865	15.00	
4	(0, 2, 1, 0, 0), (0, 1, 0, 1), (0, 0, 0), (2, 0)	400	7.00	✓
5	(0, 2, 0, 0, 0), (0, 2, 0, 1), (2, 0, 0), (2, 2)	615	11.75	✓
6	(0, 1, 0, 0, 1), (0, 0, 0, 2), (0, 2, 0), (3, 0)	515	8.25	✓
7	(1, 1, 1, 0, 0), (0, 2, 0, 2), (0, 1, 0), (2, 3)	755	12.75	✓

Now, if one considers the particular patterns selected for turbines 1, 2, and 5 in Table 2.7, the joint optimization approach suggests a combination of maintenance patterns that are more expensive but require less time to be performed. This allows for more turbines to be visited within the maintenance rotation which in turns leads to a reduction in the penalty cost, in particular, and in the overall cost in general. This clearly demonstrates the benefit of jointly solving the SMP and routing problem rather than making the decisions separately.

2.7.5 Set of experiments #4: impact of target reliability and trip length

To show how the target reliability impacts the SM plan and repair crews paths assignment, the JSMOP is solved for a limited duration $\mathcal{T}^{max} = 16$ hours while the target reliability $\mathcal{R}_0$ is varied from 96% to 99%. The results obtained are depicted in Table (2.9). This table shows the optimal number N^* of the OWTs visited, the total cost incurred $\mathcal{Z}^*$, in addition to the CPU times. From these results, one may observe that when the reliability target $\mathcal{R}_0$ increases, the number N^* of OWTs that are visited decreases while the resulting total cost $\mathcal{Z}^*$ increases.

Table 2.9: Results obtained for $\mathcal{T}^{max} = 16$ and varying values of $\mathcal{R}_0$: case of Experiments #4

$\mathcal{R}_0$(%)	N^*	$\mathcal{Z}^*$	CPUt (s)
96	7	19,650	37.5
97	6	29,774	500.8
98	6	32,934	233.3
99	3	61,227	13.4

For the particular case where $\mathcal{R}_0 = 97\%$ and $\mathcal{T}^{max} = 16$ hours, Table (2.10) lists the maintenance patterns selected for each OWT, as well as the resulting cost $\mathcal{C}_i$ ($), time $\mathcal{T}_i$ (hr), and OWT reliability $\mathcal{R}_i$ (%). In addition, for each one of the repair crews, Table (2.11) indicates the route, working time, and the required travel cost. Based on the selected repair crew routes it can be seen that the initially failed OWTs are visited as early as possible if not immediately. The three available repair crews each starts their maintenance duties by visiting a failed OWT: the first repair crew

($r = 1$) first visits OWT 7 before completing their route via OWT 4; repair crew $r = 2$ first visits the failed OWT 5, then goes to OWT 6 and returns back to the maintenance base, and the last repair crew visits OWT 3 followed by OWT 1. OWT 2 is the only one that was not selected for maintenance despite its reliability which is lower than $\mathcal{R}_0 = 97\%$. As consequence, a penalty cost will be incurred. The path selected for each repair crew is illustrated in Figure (2.3).

Table 2.10: Maintenance results obtained for $\mathcal{R}_0 = 97\%$ and $\mathcal{T}^{max} = 16$: case of Experiments #4

OWT i	Patterns selected	Cost ($)	Time (hr)	$\mathcal{R}^*$(%)
1	(0, 1, 0, 1, 0), (0, 1, 0, 0), (0, 2, 0), (0, 0)	290	4.25	97.5
2	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	–	–	85.3
3	(0, 1, 0, 0, 1), (0, 1, 0, 2), (0, 2, 0), (2, 2)	565	9.75	97.1
4	(0, 0, 1, 0, 0), (0, 1, 0, 1), (0, 0, 0), (0, 0)	200	2.75	97.2
5	(0, 1, 0, 0, 0), (0, 1, 0, 1), (0, 0, 0), (2, 1)	350	6.00	97.6
6	(0, 1, 0, 0, 1), (0, 0, 0, 0), (0, 1, 0), (2, 0)	280	4.50	97.1
7	(1, 1, 1, 0, 0), (0, 1, 0, 0), (0, 1, 0), (1, 3)	580	9.00	97.4

Table 2.11: Repair crew assignment and routing for $\mathcal{R}_0 = 97\%$ and $\mathcal{T}^{max} = 16$: case of Experiments #4

Repair crews	Route	Working time (hr)	Travel cost ($)
RC1	[0, 7, 4, 0]	13.27	56.92
RC2	[0, 5, 6, 0]	12.05	58.05
RC3	[0, 3, 1, 0]	15.13	42.31

Now, to investigate the impact of the limited duration $\mathcal{T}^{max}$ allotted to the maintenance rotation on the SM plan and repair crews routes assignment, the JSMOP is solved for a required minimum reliability $\mathcal{R}_0 = 98\%$ and varied values of the shift duration $\mathcal{T}^{max}$. The benefit of such an investigation can be justified from the fact that, in real-world wind farm maintenance setting, shift lengths are frequently shortened or interrupted due to operating environment conditions (e.g., poor weather conditions).

The results obtained for values of $\mathcal{T}^{max} \in \{14, 16, 18, 20\}$ are reported in Table (2.12). For each value of $\mathcal{T}^{max}$, this table gives the optimal number N^* of the OWT

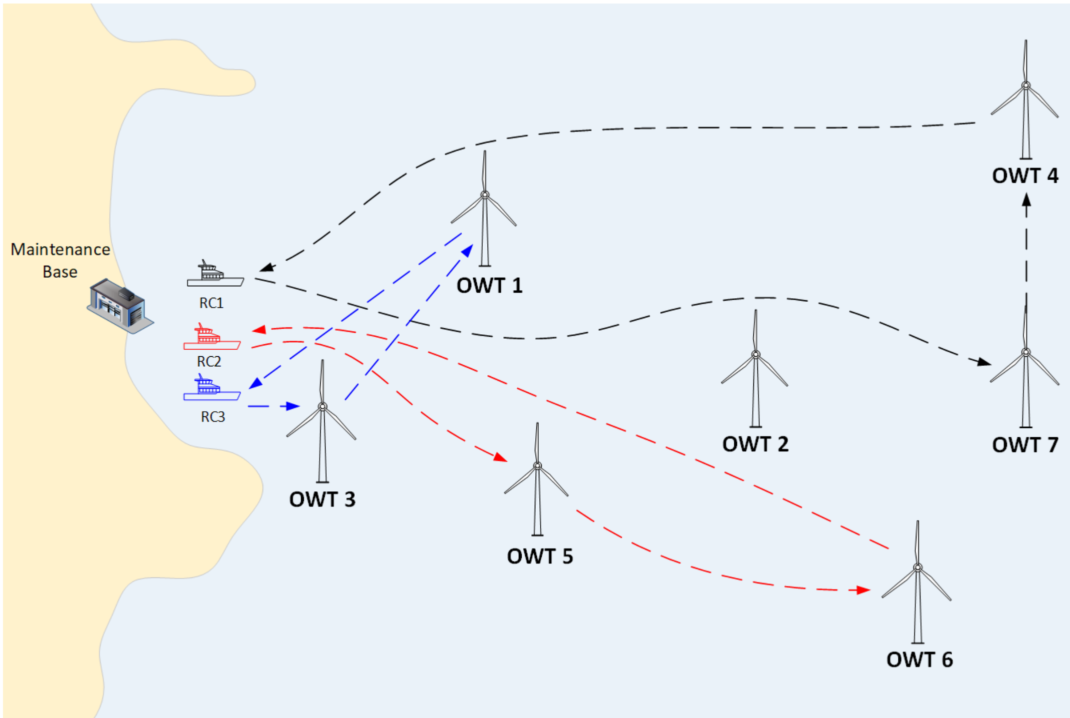

Figure 2.3: Illustration of paths selected for each repair crew

to visit, the total cost $\mathcal{Z}^*$ incurred, in addition to the CPU time. From these results, it is observed that as the shift length is shortened the optimal number N^* of visited OWTs decreases while the total cost $\mathcal{Z}^*$ increases.

Table 2.12: Results when the shift length is varied: case of Experiments #4

$\mathcal{T}^{max}$	Visited nodes	$\mathcal{Z}^*$	CPUt (s)
14	5	41,345	237.5
16	6	32,934	233.3
18	6	31,818	356.7
20	7	23,412	78.2

2.7.6 Set of experiments #5: selecting the optimal number of repair crews

For this set of experiments, the problem described in experiment #2 is extended to include $R = 3$ potential repair crews with different skill levels and costs. For each repair crew $r \in \{1, 2, 3\}$, Table (2.13) shows the repair crews speed coefficients ϵ_r, the variable cost c_r^v and the fixed cost c_r^f. From these data, repair crew 1 ($r = 1$) is characterized by speed coefficient $\epsilon_1 = 0.5$ which means that the repair crew is capable of carrying out a maintenance action in half the required standard duration. With a speed coefficient $\epsilon_2 = 1$, the second repair crew has standard/baseline qualification. The third repair crew is the least skilled with a speed coefficient $\epsilon_2 = 1.2$ which means that it requires 20% more time than repair crew 2 to perform the same maintenance action.

Table 2.13: Repair crew data: case of Experiments #5

Repair crew r	ϵ_r	c_r^v	c_r^f
1	0.50	40	275
2	1.00	30	200
3	1.20	20	125

The JSMOP is solved for a limited shift duration $\mathcal{T}^{max} = 15$ hours while the required minimum reliability $\mathcal{R}_0$ varies from 86% to 99%. The optimal results obtained, namely the number N^* of visited OWTs, the total cost incurred $\mathcal{Z}^*$, the repair crews used, their routes and the CPU times are depicted in Table (2.14). From this table, the following observations can be made:

- When the required minimum reliability $\mathcal{R}_0 = 86\%$, repair crew #1 (the fastest crew) performs 15 maintenance actions on all 7 turbines for a total maintenance time of 7.125 hours. According to the detailed experiment results in Appendix B, the 15 maintenance actions are spread as follows: 14 minimal repairs (MR) and one imperfect maintenance (IM).

- When $\mathcal{R}_0$ increases to 90%, repair crew #1 is no longer capable of visiting all turbines as more maintenance activities are required meaning spending more time at the turbines. Therefore, the model adds repair crew #3 to perform the maintenance actions required at turbine 5. In total, 19 maintenance actions (18 MR and one IM) are conducted and take 11.225 hours (See Appendix B).

- When $\mathcal{R}_0$ is further increased to 94%, repair crews #1 and #3 are no longer

Table 2.14: JSMOP results for varying values of target reliability: case of Experiments #5

$\mathcal{R}_0$	N	Crews used #1	#2	#3	Routes	m	$\mathcal{T}_m$	$\mathcal{Z}^*$ (\$)	CPUt(s)
86%	7	✓			[0,3,5,7,4,6,2,1,0]	15	7.125	13,555	94.3
90%	7	✓			[0,3,7,4,6,2,1,0]	16	7.625	14,380	79.7
				✓	[0,5,0]	3	3.600		
94%	7	✓			[0,3,7,4,6,2,1,0]	22	12.125	16,335	112.7
			✓		[0,5,0]	4	4.000		
98%	6	✓			[0,7,4,2,1,0]	20	12.875	30,059	387.6
			✓		[0,3,0]	7	11.750		
				✓	[0,5,0]	5	8.700		
99%	4	✓			[0,7,6,0]	13	11.625	52,041	87.7
			✓		[0,5,0]	6	12.250		
				✓	[0,4,0]	5	8.700		

m: number of maintenance actions carried out by repair crew
$\mathcal{T}_m$: duration in hours of maintenance actions carried out by repair crew

capable of visiting all turbines as more high level maintenance activities are required meaning spending more time at the turbines. Therefore, the model replaces repair crew #3 with the faster repair crew #2 to perform the maintenance actions required at turbine 5. In total, 26 maintenance actions (23 MR, two IM, and one Replacement) are conducted and take 16.125 hours. It should be noted that the number of maintenance actions performed increases as well as their quality (level): two IM and one replacement are carried out.

- When $\mathcal{R}_0$ reaches to 98%, all three repair crews are used. However, the reliability requirement is so high that there is not enough time to perform the required maintenance actions needed. Thus, turbine 6 is not visited hence incurring a penalty cost. In this case, 32 maintenance actions (21 MR, eight IM, and three replacements) are performed for a total maintenance time of 33.325 hours. Similar model behavior is observed for $\mathcal{R}_0 = 99\%$. The reliability requirement far exceeds the capacity of the repair crews for the maintenance rotation time available. Thus, turbines 1, 2, and 3 are not visited and turbine 3 remains down. In this case, 24 maintenance actions (9 MR, 12 IM, and three replacements) are performed for a total maintenance time of 32.575 hours.

- From these results, one can conclude that as the target reliability is increased the number of repair crews utilized is increased because it takes longer to carry out the maintenance actions needed to guarantee increasing reliability. In general, the computation time also increases as the reliability requirements are increased. When the repair crew capacity is reached, the number of turbines visited drops and penalties are incurred.

The sets of experiments carried out above clearly show that the proposed joint formulation of the SMP and OP is efficient and yields valid maintenance and routing decisions. The joint formulation is superior to the sequential approach without adding significant computation burden for small and moderately large systems.

2.8 Conclusion

This chapter introduced a mathematical formulation to optimally solve the joint selective maintenance and orienteering problem. The pattern generation algorithm developed by Diallo et al. (2018) was extended to generate all feasible maintenance patterns and eliminate the need for solving a nonlinear formulation. The proposed model was applied to a real-world problem of selecting the maintenance actions to be performed on a set of offshore wind turbines and also determining the optimal route for several repair crews. Multiple numerical experiments were run to show that the joint model produces cheaper overall solutions than the traditional approach of sequentially solving the SMP followed by the resulting routing problem. The experiments also demonstrated key characteristics and properties of the model.

Future extensions of the work produced in this chapter include eliminating some of the assumptions made regarding the way in which maintenance operations are conducted at offshore wind farms. The assumptions that service vessels are required to be present while maintenance is being performed and that they can carry only a single repair crew is not necessarily realistic. It may also be somewhat unrealistic to assume that service vessels have the necessary capacity to carry any spare parts/tools that may be required for maintenance. Developing a model that drops the described assumptions may provide a more accurate and realistic representation of the problem.

Reliability was used as the desired performance indicator for turbines, however availability is another common indicator that is often used in practice. An innovative and novel extension of the SMP and joint JSMOP could be to develop an optimization model with an objective of selecting a subset of components and maintenance actions to be performed on the selected components in which the system availability over the upcoming mission (mean availability) meets or exceeds an availability target. The mean system availability refers to the percentage of time that a system is operable over a mission length and in many industrial applications would be of more importance than the system reliability. However, availability is much more complex performance indicator than system reliability.

One last area of future research is the solving of the JSMOP for very large instances of the problems with hundreds of systems comprised of hundreds of components. This would require the development of decomposition methods, the use of column generation to reduce the number of patterns generated, and matheuristics based on recent developments in the solution of the vehicle routing problem with profits.

Chapter 3

A column generation-based approach for solving the multimission selective maintenance optimization problem in large-scale serial k-out-of-n:G systems

3.1 Introduction

Many military and industrial systems are required to perform a series of missions with finite breaks between missions. Examples of these systems include aerospace equipment, weapon systems, computers, production and manufacturing lines, and energy production systems. To maintain such systems and ensure they continue to operate their missions at an acceptable performance level, maintenance actions usually need to be carried out during scheduled breaks. Due to resource constraints such as time, cost, and repair-person availability, only a subset of system components can be selected for maintenance operations. Such a maintenance decision problem is known in the literature as the selective maintenance problem (SMP). The SMP aims to identify the optimal subset of components to maintain and the level of maintenance actions to be performed on the selected components so that to meet a specified system performance for the upcoming missions.

The original selective maintenance model was first introduced by Rice et al. Rice et al. (1998) and considered a series-parallel system where components lifetimes are exponentially distributed. Accordingly, the only maintenance action considered is perfect repair of failed components. The objective of the proposed model was to maximize system reliability for the next mission and an enumeration method was used to find the optimal SM plan. In the intervening years since Rice et al. (1998) proposed the first selective maintenance model, many researchers have expanded upon this work. These studies have included complex system configurations Cassady et al. (2001b); Diallo et al. (2018), multistate systems Liu and Huang (2010); Pandey et al.

(2013a), component dependence Xu et al. (2016); Dao and Zuo (2017), fleet level selective maintenance Khatab et al. (2020); Schneider and Cassady (2015), stochastic break and/or mission duration Liu et al. (2018); Khatab et al. (2017), condition-based SMP Khatab et al. (2018a), and multiple repair channels Diallo et al. (2019); Khatab et al. (2018b). A literature review of the SMP is provided in Xu et al. (2015). A more recent SMP literature review was conducted by Cao et al. (2018).

The majority of the existing models of the SMP deals with the single mission case. However in real industrial situations, it is often desirable to obtain SM plans for a time horizon composed of a sequence of alternating missions and breaks. In such situations, there are clear and strong trade-offs between the maintenance decisions and long-term resources management. Furthermore, from an optimization point of view, an optimal global maintenance plan covering all missions at once is better than sequentially planning the maintenance for multiple single missions separately. This extended SM decision problem is referred to as the multimission selective maintenance problem (MMSMP). The resulting combinatorial optimization problem (MMSMOP) includes a third maintenance decision: selecting the appropriate break during which a maintenance action will take place. The MMSMOP is usually formulated as a mixed integer nonlinear problem due to the system reliability. Its complexity increases exponentially as the number of system components and missions increases.

A limited number of papers dealing with the MMSMP have been published in the literature. Khatab and Ait-Kadi (2008) develop a SMP optimization model in multistate systems (MSS) operating several missions. The resulting optimization program is solved using a simulated annealing algorithm. Maillart et al. (2009) analyze the finite and infinite-horizon multimission SMP for series parallel systems. Both the finite and infinite-horizon problems are formulated as stochastic dynamic programs. Through numerical experiments it is shown that there is rarely a difference between the optimal finite and infinite-horizon SM policies. Zhao and Zeng (2016) propose a model for the multimission SMP where the break duration is exponentially distributed. Pandey et al. (2016) investigate the finite-horizon MMSMP for a series parallel system and provide a nonlinear programming formulation. In their work, the hybrid imperfect maintenance model is used to model the imperfect maintenance options. Zhang et al. (2019) extend the work in Pandey et al. (2016) to a MSS whose components deteriorate according to an homogeneous-time Markov chain. The authors develop a SM optimization model to determine the maintenance times and options for each system component to minimize the total maintenance cost under system reliability requirement.

Recent papers (Jiang and Liu, 2020b; Chaabane et al., 2020; Shahraki et al., 2020) addressed the MMSMP. Jiang and Liu (2020b) extend the results in (Khatab et al., 2017) to deal with the MMSMP with uncertain mission duration and component effective ages. A max-min optimization model is presented and solved using a simulated annealing-based genetic algorithm. Chaabane et al. (2020) investigate the combined multimission selective maintenance and repair-person assignment problem. A mixed integer nonlinear programming model is proposed and solved using a genetic algorithm. Shahraki et al. (2020) also investigate the multimission SMP in MSS where components are subjected to s-dependency and random IM. The s-dependency between components is represented by two types of interactions as a function of the system performance rate in addition to the number of components impacted. A multi-objective SMOP is then developed to jointly maximize the system reliability and minimize its corresponding variability.

The majority of the papers discussed above are still based on elementary series-parallel systems structures. However, more complex structures such as k-out-of-n:G can be encountered in a wide range of industrial applications including electronics, telecommunication networks, and power generation (Elsayed, 2012). A k-out-of-n:G structure, also denoted as GA(k, n), consists of n components and operates if at least k components operate. It generalizes the series-parallel structure as it refers to a parallel structure when $k = 1$, while it reduces to a series structure when $k = n$. To our knowledge, Diallo et al. (2018) is the only paper that deals with the SMP in serial GA(k, n) systems. The authors in (Diallo et al., 2018) propose two new nonlinear SMP formulations. To solve the resulting SM optimization problems, a two-phase approach using a binary integer programming (BIP) model is developed: the first phase generates all feasible maintenance patterns for each GA(k, n) subsystem, while the second phase solves a multidimensional multiple-choice knapsack problem (MMKP) to select the optimal mix of maintenance patterns.

The SM approach proposed in Diallo et al. (2018) considers a serial GA(k, n) systems operating only one subsequent mission. The present chapter develops a new approach where the SMP in Diallo et al. (2018) is comprehensively extended to help maintenance decision makers resolve real industry occurrences of the SMP in serial GA(k, n) systems operating several missions interspersed by scheduled breaks with possibly different lengths. The lifetimes of system components are generally distributed and subjected to a list of several IM levels including replacement. To meet a minimum required performance level for the next mission, the maintenance activities are performed on the system components during the break. To avoid unplanned interruptions due to components failures during a mission, it is assumed that minimal repair can be carried out on failed components. This assumption has been already adopted in the literature (Chaabane et al., 2020; Khatab and Ait-Kadi, 2008) since it is not uncommon to find systems where a subsystem is composed of only one component or machine. Such a system is indeed investigated in Zhu et al. (2011) as a machining line of the automobile engine connecting rod. The machining line is composed of 10 subsystems in series where the first subsystem is a single machine. Failure of the single machine would stop the production line and cause economic losses. Due to the limited duration allotted to the scheduled breaks, not all components are likely to be maintained. The objective is to find the optimal decisions minimizing the total maintenance cost while ensuring a minimum reliability level during the missions. The combinatorial complexity of the resulting optimization model is naturally higher than that in Diallo et al. (2018). Therefore, there is a need to develop effective approaches in solving such complicated non-linear optimization problems.

The objectives of the present chapter are twofold. We first propose an extension of the SMP in serial GA(k, n) systems under a multimission planing horizon. Given the high complexity inherent to the resulting MMSMOP, a new solution method is developed while combining the column-generation (CG) approach with the genetic algorithm (GA). Column-generation is an efficient method for solving large linear programs dealing with a huge number of variables. It was first used by Gilmore and Gomory (1961) and Gilmore and Gomory (1963) to solve the cutting stock problem and has since been applied to a wide range of optimization problems. These problems include, for example, the redundancy allocation problem (Zia and Coit, 2010), vehicle routing problem (Faiz et al., 2019), shift scheduling (Al-Yakoob and Sherali, 2008), and dynamic job assignment (Range et al., 2019). Liu et al. (2010) also apply CG to the passenger rail crew scheduling, and used a GA to solve the induced subproblems. Dunbar et al. (2020) propose a CG approach where the restricted master problem is solved exactly and a GA is used to solve the subproblems. The results of this study indicate that the approach yields improved solutions compared to the current best-case costs.

Column-generation has yet to be applied to any variant of the SMP. In the present work, a hybrid CG and GA is proposed for the MMSMP to find efficient and high quality SM plans. Comprehensive experiments and comparison with other proposed solution methods highlight better performance of the proposed hybrid solution approach. For large scale MMSMP, it will be shown that the proposed solution approach is capable not only of obtaining near optimal solutions but also to outperform other solution approaches that have been used in the literature. Furthermore, the experiments conducted will show that the present work will lead to efficient solution methods for more complex SM problems.

The remainder of this chapter is structured around six additional sections. Section 3.2 presents the system of notation used and the main working assumptions made in the present work. This section also describes the system investigated and the associated reliability computations. Section 3.3 presents the imperfect maintenance model, in addition to the computation of total maintenance time and cost. Section 3.4 presents the MINLP formulation and the corresponding BIP formulation of the MMSMP. The proposed hybrid solution approach combining CG and GA is developed in Section 3.5. This section also presents the formulation of the restricted master problem and subproblems. Section 3.6 presents several sets of numerical experiments carried out to validate the proposed SM modeling and solution approach. Conclusions and future research extensions are presented in section 3.7.

3.2 System description and reliability computation

The system of notation and main working assumptions used in the present chapter are presented below.

3.2.1 Notation

$\mathcal{I}$	Set of subsystems, $\mathcal{I} = \{1, 2, ..., N\}$ with index i
$\mathcal{J}_i$	Set of components in subsystem i, $\mathcal{J}_i = \{1, 2, ..., N_i\}$ with index j
$\mathcal{M}$	Set of missions, $\mathcal{M} = \{1, 2, ..., M\}$ with index m
$\mathcal{L}_{ij}$	Set of preventive maintenance levels available for component E_{ij}, $\mathcal{L}_{ij} = \{0, 1, ..., L_{ij}\}$ with index l
$\mathcal{P}_i$	Set of maintenance patterns generated for subsystem i, $\mathcal{P}_i = \{1, 2, ..., P_i\}$ with index p
K_i	Minimum number of components that must be functioning in subsystem i
E_{ij}	The j^{th} component of subsystem i
X_{ijm}	Age of component E_{ij} at the start of break m
Y_{ijm}	Age of component E_{ij} at the end of break m
$h_{ij}(t)$	Failure rate of component E_{ij}
t_{ijl}	Duration of PM level l on component E_{ij}
c_{ijl}	Cost of PM level l on component E_{ij}
c^r_{ij}	Cost of minimal repair on component E_{ij}
U_m	Duration of mission m
D_m	Duration of break m
$\mathcal{R}^c_{ij}$	Conditional reliability of component E_{ij} during mission m
$\mathcal{R}^s_{im}$	Reliability of subsystem i during mission m
$\mathcal{R}_m$	Overall system reliability during mission m
$\mathcal{R}_{0m}$	Minimum required reliability level during mission m

3.2.2 Main working assumptions

1. The system is comprised of multiple subsystems aranged in a series configuration. Each subsystem is made up of multiple binary components (components and system can either be functioning or failed).

2. During a break, system components do not age, i.e. the age of a component is operation-dependent.

3. If a component fails during a mission, a minimal repair is performed to bring the component back to the functioning state. When a minimal repair is performed on a failed component, its failure rate remains undisturbed. Failures occur following a non-homogeneous Poisson process (NHPP).

4. The time to perform a minimal repair is negligible compared to the mission duration.

5. All required limited resources (budget, repair-persons, tools) are available when needed. Only one repair channel is available meaning that only one component can be worked on at any given time.

3.2.3 System description

The multimission SMP addressed in the present chapter considers a system comprised of N subsystems arranged in a series configuration. The i^{th} subsystem $(i = 1, \cdots, N)$ is represented by a K_i-out-of-N_i:G reliability bloc diagram (RBD). In reliability theory, the K_i-out-of-N_i:G structure is usually denoted as GA(K_i, N_i) and specifies that the system is functioning if and only if at least K_i out of the N_i components are functioning. Such RBD is a generalization of both the series (case of $K_i = N_i$) and parallel (case of $K_i = 1$) structures. Individual components in each subsystem are independent and their lifetimes are not necessarily identically distributed.

The system under consideration is required to perform a series of missions each separated by scheduled maintenance breaks of finite length. It is assumed that the system has just completed a mission and is entering the first break of a new sequence of missions. There are M subsequent missions and as many scheduled maintenance breaks indexed by m ($m \in \{1, 2, \cdots M\}$). At the end of mission m the system is switched off for the $(m+1)$th break of duration D_{m+1} during which maintenance actions can be performed. The system will be required to operate the following mission of duration U_{m+1} at a required minimum reliability level $\mathcal{R}_{0,m+1}$.

Let X_{ijm} and Y_{ijm} denote the effective age of component E_{ij} at the start and end of maintenance break m, respectively. All components are subjected to perfect inspections that reveal their effective ages X_{ij1} at the start of the first break. The duration of these perfect inspections are not included in the break duration. The recursive relationship between X_{ijm} and Y_{ijm} is given by the following equation:

$$X_{ij,m+1} = Y_{ijm} + U_m, \quad \forall m \in \mathcal{M} \setminus \{M\}. \tag{3.1}$$

3.2.4 Reliability computation

It is required that the system successfully operates all missions at a predetermined minimum required reliability level. The system reliability for a particular mission m is defined by the probability that it can successfully complete the mission. To compute the system reliability for a particular mission, one must first compute $\mathcal{R}^c_{ijm}$ and $\mathcal{R}^s_{im}$, the component and subsystem reliabilities, respectively. Given that component E_{ij} has effective age Y_{ijm} at the end of maintenance break m (i.e at the start of mission m), the probability that it will survive the next mission of duration U_m is defined by the conditional reliability function:

$$\mathcal{R}^c_{ijm} = \frac{\mathcal{R}_{ij}(Y_{ij} + U_m)}{\mathcal{R}_{ij}(Y_{ij})}, \tag{3.2}$$

where $\mathcal{R}_{ij}(t)$ refers to the unconditional reliability of component E_{ij}.

Without loss of generality, it is assumed that component E_{ij} has a Weibull distributed lifetime with shape and scale parameters β_{ij} and η_{ij}, respectively. In this case, the probability $\mathcal{R}^c_{ijm}$ of component E_{ij} to successfully operate mission m becomes:

$$\mathcal{R}^c_{ijm} = \exp\left[\left(\frac{Y_{ijm}}{\eta_{ij}}\right)^{\beta_{ij}} - \left(\frac{Y_{ij}+U_m}{\eta_{ij}}\right)^{\beta_{ij}}\right]. \tag{3.3}$$

The reliability $\mathcal{R}^s_{im}$ of the the i^{th} subsystem is obtained from the exact formulation proposed in Arulmozhi (2002):

$$\mathcal{R}^s_{im} = \sum_{j_{k_i}=1}^{N_i} \sum_{j_{k_i-1}=1}^{j_{k_i}-1} \cdots \sum_{j_1=1}^{j_2-1} \left(\prod_{v=j_1}^{j_{k_i}} \mathcal{R}^c_{iv}\right) \left(\prod_{\substack{u=1 \\ u\neq j_1,\cdots,j_{k_i}}}^{j_{k_i}} (1-\mathcal{R}^c_{iu})\right). \tag{3.4}$$

To compute the reliability $\mathcal{R}^s_{im}$ of each subsystem, the algorithm proposed by Kuo and Zuo (2003) is implemented. Because subsystems are arranged in a series configuration, the overall system reliability for mission m is then computed as:

$$\mathcal{R}_m = \prod_{i\in I} \mathcal{R}^s_{im}. \tag{3.5}$$

3.3 Imperfect maintenance modeling, and total maintenance time and cost computation

For each component E_{ij}, there is a list $\mathcal{L}_{ij} = \{0, \cdots, L_{ij}\}$ of $L_{ij}+1$ preventive maintenance levels $l \in \mathcal{L}_{ij}$ that can be selected during breaks. These maintenance levels include do-nothing, imperfect maintenance (IM), and replacement. The do-nothing ($l = 0$) case refers to no maintenance being performed on the component. The replacement level ($l = L_{ij}$) resets the component's age to 0, while an IM level $0 < l < L_{ij}$ if selected will return the component's age between that obtained after minimal repair and replacement. When carried out on a component E_{ij}, a PM of level l requires t_{ijl} time units, and costs c_{ijl} monetary units.

Commonly used IM models in the literature are: age reduction (Malik, 1979), hazard rate adjustment (Nakagawa, 1988), and hybrid hazard rate (Lin et al., 2000) models. The age reduction approach is adopted here to model IM. Accordingly, each

IM level $l \in \mathcal{L}_{ij}$ available for component E_{ij} is characterized by an age reduction coefficient γ_{ijl} $(0 \leq \gamma_{ijl} \leq 1)$. Therefore, when the IM is performed on E_{ij}, its effective age X_{ijm} is reduced and becomes:

$$Y_{ijm} = \gamma_{ijl} \cdot X_{ijm}. \tag{3.6}$$

For modeling purpose, the following decision variable w_{ijlm} is introduced as:

$$w_{ijlm} = \begin{cases} 1, & \text{if maintenance level } l \text{ is performed} \\ & \text{on } E_{ij} \text{ during break } m, \\ 0, & \text{otherwise.} \end{cases} \tag{3.7}$$

Using Equation (3.7), the total time $\mathcal{T}_m$ spent performing maintenance actions during break m is expressed as:

$$\mathcal{T}_m = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} \sum_{l \in \mathcal{L}_{ij}} t_{ijl} \cdot w_{ijlm}. \tag{3.8}$$

According to assumption 3, if component E_{ij} fails during a mission, a minimal repair is performed at a cost of c^r_{ij} monetary units. Thus, the total expected maintenance cost $\mathcal{C}$ can be expressed as:

$$\mathcal{C} = \left(\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} \sum_{l \in \mathcal{L}_{ij}} \sum_{m \in \mathcal{M}} c_{ijl} \cdot w_{ijlm} + c^r_{ij} \cdot \int_{Y_{ijm}}^{Y_{ijm}+U_m} h_{ij}(t)\, dt \right), \tag{3.9}$$

where $h_{ij}(t)$ is the failure rate of component E_{ij}, the first term represents the cost of the PM actions and the second term represents the expected minimal repair cost.

3.4 Mathematical programming formulations

Before establishing the mathematical programming formulations of the multimission SMP (MMSMP), let us recall that the system is designed to operate the missions at a predetermined minimum required reliability level. The goal of the MMSMP is to jointly select the set of components to be maintained in each break, and the maintenance levels to be performed on the selected components to minimize the grand total maintenance cost subjected to the minimum required reliability level during each subsequent mission $\mathcal{R}_{0m}$. Thus, the maintenance actions selected must result in the system meeting or exceeding the minimum reliability for each mission. In this section we first introduce the mixed integer nonlinear programming formulation of the MMSMP. The proposed MINLP formulation is an adaptation of the one presented in (Chaabane et al., 2020) as, in the present work, only a single repair crew is accounted for. Then, a binary integer programming (BIP) formulation of the MMSMP is proposed based on a pattern enumeration approach. The objective of both proposed optimization models is to minimize the total expected cost while meeting the required minimum reliability level for each mission.

3.4.1 MINLP formulation of the multimission SMP

The proposed MINLP formulation of the MMSMP is:

$$\min_{w_{ijlm}\in\{0,1\}} \quad \left(\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}_i}\sum_{l\in\mathcal{L}_{ij}}\sum_{m\in\mathcal{M}} c_{ijl}\cdot w_{ijlm} + c^r_{ij}\cdot\int_{Y_{ijm}}^{Y_{ijm}+U_m} h_{ij}(t)\,dt\right) \tag{3.10}$$

subject to:

$$\prod_{i\in\mathcal{I}}\mathcal{R}^s_{im}\geq\mathcal{R}_{0m},\quad \forall m\in\mathcal{M} \tag{3.11}$$

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}_i}\sum_{l\in\mathcal{L}_{ij}} t_{ijl}\cdot w_{ijlm}\leq D_m,\quad \forall m\in\mathcal{M} \tag{3.12}$$

$$Y_{ijm}=X_{ijm}\cdot\left(\sum_{l\in\mathcal{L}_{ij}}\gamma_{ijl}\cdot w_{ijlm}\right),\quad \forall j\in\mathcal{J}_i,\forall m\in\mathcal{M} \tag{3.13}$$

$$X_{ijm+1}=Y_{ijm}+U_m,\quad \forall i\in\mathcal{I},\forall j\in\mathcal{J}_i,\forall m\in\mathcal{M}\setminus\{M\} \tag{3.14}$$

$$\sum_{l\in\mathcal{L}_{ij}} w_{ijlm}=1,\quad \forall i\in\mathcal{I},\forall j\in\mathcal{J}_i,\forall m\in\mathcal{M} \tag{3.15}$$

$$\mathcal{R}^c_{ijm}=\frac{\mathcal{R}_{ij}(Y_{ij}+U_m)}{\mathcal{R}_{ij}(Y_{ij})},\quad \forall i\in\mathcal{I},\forall j\in\mathcal{J}_i,\forall m\in\mathcal{M} \tag{3.16}$$

In the above optimization model, the objective function (3.10) minimizes the grand total expected maintenance cost. Constraints (3.11) and (3.12) are the minimum required reliability level for each mission and breaks duration constraints respectively. Constraints (3.13) update the effective age of the components at the end of breaks, while constraints (3.14) are recurrence relations computing the components ages at the end of each mission. For each component, Constraints (3.15) ensure that a single maintenance level is selected for each component. Constraints (3.16) are used to compute the components conditional reliability functions.

3.4.2 BIP formulation with complete subsystem pattern information

In this section, an alternative formulation to the MINLP previously discussed is presented. This formulation relies on a full enumeration of all maintenance patterns $\mathcal{P}_i$ for each subsystem $i \in \mathcal{I}$. A maintenance pattern $p \in \mathcal{P}_i$ can be defined as a combination of components and related maintenance levels to be performed during each break. Accordingly, for each pattern $p \in \mathcal{P}_i$ corresponds a total expected cost $\mathcal{C}_{ip}$ and time $\mathcal{T}_{imp}$ to perform the selected maintenance actions during the m^{th} break, as well as reliability $\mathcal{R}^s_{imp}$ of subsystem i for each mission $m \in \mathcal{M}$. A pattern $p \in \mathcal{P}_i$ is then represented as a column-vector of $N_i \times M$ elements whose values are the maintenance levels performed on the components of the i^{th} subsystem. In a pattern $p \in \mathcal{P}_i$, the first group of N_i elements represents the maintenance levels performed during the first ($m = 1$) break, the second group of N_i elements denote the maintenance levels carried out during the second ($m = 2$) break, and so on.

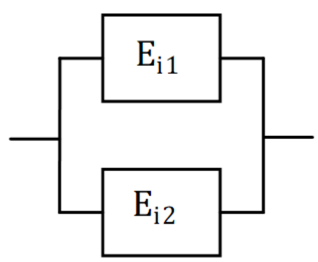

Figure 3.1: Parallel subsystem

To illustrate the generation of maintenance patterns, let us consider a GA(1, 2) system as shown in Figure 3.1 with two levels of maintenance: Do nothing ($l = 0$) and replacement ($l = 1$). If the system is required to perform a sequence of two missions interspersed by two breaks, then all 16 possible maintenance patterns are:

$$\begin{bmatrix} 0 \\ 0 \\ - \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 1 \\ - \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 1 \\ - \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 0 \\ - \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 1 \\ - \\ 1 \\ 0 \end{bmatrix}, \cdots, \quad \begin{bmatrix} 1 \\ 1 \\ - \\ 1 \\ 1 \end{bmatrix}$$

The first pattern (column-vector) means that no maintenance is performed on both components during both breaks. The second pattern means that only component E_{12} is replaced during the first break. According to the third pattern, components E_{11} andE_{12} are replaced during the first maintenance break, while no maintenance action is performed during the second break. The fourth pattern would mean that only component E_{11} is replaced during the first break, and only component E_{12} is replaced during the second break.

Now, assuming that for each subsystem $i \in \mathcal{I}$ complete pattern information $\mathcal{P}_i$ is available, the proposed BIP formulation of the multimission SMP with the objective of minimizing total expected cost is written as:

$$\min_{z_{ip}} \quad \sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}_i} C_{ip} \cdot z_{ip} \tag{3.17}$$

Subject to:

$$\sum_{p \in \mathcal{P}_i} z_{ip} = 1, \quad \forall i \in \mathcal{I} \tag{3.18}$$

$$\sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}_i} \mathcal{T}_{imp} \cdot z_{ip} \leq D_m \quad \forall m \in \mathcal{M} \tag{3.19}$$

$$\sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}_i} \ln(\mathcal{R}^s_{imp}) \cdot z_{ip} \geq \ln(\mathcal{R}_{0m}) \quad \forall m \in \mathcal{M} \tag{3.20}$$

$$z_{ip} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, p \in \mathcal{P}_i \tag{3.21}$$

In the above optimization model, constraints (3.18) ensure that a single maintenance pattern is selected for each subsystem i. Constraints (3.19) ensure that the maintenance time during break m does not exceed the available working time. Constraints (3.20) guarantee that the system reliability during mission m must be greater than or equal to the target reliability $\mathcal{R}_{0m}$. The term on the left-hand side of the inequality is the linearization of the nonlinear systems reliability function of Equation (3.5). This linearization is obtained by the following procedure:

$$\begin{aligned}\mathcal{R}_m &= \prod_{i\in\mathcal{I}} \mathcal{R}^s_{im} \\ &= \prod_{i\in\mathcal{I}} \left(\sum_{p\in\mathcal{P}_i} \mathcal{R}^s_{imp} \cdot z_{ip} \right).\end{aligned} \tag{3.22}$$

$$\mathcal{R}_m = \prod_{i\in I} \left(\sum_{p\in P_i} \mathcal{R}^s_{imp} \cdot z_{ip} \right) \tag{3.23}$$

Because z_{ip} is a binary variable together with the monotonicity of the natural logarithm function, the result in constraint (3.20) is straightforward obtained as:

$$\ln(\mathcal{R}_m) = \sum_{i\in\mathcal{I}} \sum_{p\in\mathcal{P}_i} \ln\left(\mathcal{R}^s_{imp}\right) \cdot z_{ip} \tag{3.24}$$

It is worth noting that the full enumeration of all patterns quickly becomes computationally inefficient for systems of even moderate size operating multiple missions. To overcome this drawback, one may resort to the column-generation (CG) approach which is discussed in the following section. The CG approach is based on a restricted master problem that is initialized with a small set of patterns, and multiple sub-problems that are used to generate promising maintenance patterns for each subsystem.

3.5 Column-generation approach

As pointed out above, the major shortcoming of the BIP formulation is that it relies on all feasible maintenance patterns being generated at the outset. There is a finite, yet extremely large number of feasible patterns that exist for systems of even moderate size. Generating all patterns up front is not realistic due to storage and CPU time limitations. This section will describe the CG algorithm that has been developed to solve the multimission SMP. To apply CG, the BIP given by Equation (3.17-3.21) in section 3.4.2 is decomposed into a restricted master problem (RMP) and multiple subproblems also referred to as pricing subproblems. The RMP is a relaxation of the BIP formulation that starts with only a small subset of feasible maintenance patterns. The subproblems are solved to generate new maintenance patterns for each subsystem that are then added to the RMP. Applying CG to the multimission SMP allows us to generate maintenance patterns iteratively and only add patterns that are promising to the RMP. Here, promising patterns refer to patterns that have negative reduced cost.

3.5.1 Restricted master problem

The restricted master problem is formulated as follows:

$$\min_{0 \leq z_{ip} \leq 1} \quad \sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}'_i} C_{ip} \cdot z_{ip} \tag{3.25}$$

subject to:

$$\sum_{p \in \mathcal{P}'_i} z_{ip} = 1, \quad \forall i \in \mathcal{I} \qquad (\theta_i) \tag{3.26}$$

$$\sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}'_i} \mathcal{T}_{imp} \cdot z_{ip} \leq D_m \quad \forall m \in \mathcal{M} \qquad (\pi_m) \tag{3.27}$$

$$\sum_{i \in \mathcal{I}} \sum_{p \in \mathcal{P}'_i} \ln(\mathcal{R}^s_{imp}) \cdot z_{ip} \geq \ln(\mathcal{R}_{0m}) \quad \forall m \in \mathcal{M} \qquad (\lambda_m) \tag{3.28}$$

The formulation presented is almost identical to the BIP formulation, however the variable z_{ip} has been relaxed and only a subset ($\mathcal{P}'_i \subset \mathcal{P}_i$) of maintenance patterns are considered initially. By solving the RMP, the dual variables $(\theta_i, \pi_m, \lambda_m)$ can be obtained and used in the pricing subproblems to identify whether there are any columns or maintenance patterns that should be added to the set $\mathcal{P}'_i$. The initial set of feasible solutions is generated using a simple heuristic of randomly assigning maintenance actions to be performed on components.

3.5.2 Pricing subproblems

To each subsystem $i \in \mathcal{I}$ corresponds a CG sub-problem. Therefore, there are N column generating subproblems that are used to find promising maintenance patterns. According to Lübbecke and Desrosiers (2005), the column with the most negative reduced cost should be added to the RMP. To identify the column that minimizes the reduced cost for each subsystem the following optimization problem is solved:

$$\min_{w_{ijlm} \in \{0,1\}} \quad C_i - \sum_{m \in \mathcal{M}} \ln(\mathcal{R}^s_{im}) \cdot \lambda_m - \sum_{m \in \mathcal{M}} T_{im} \cdot \pi_m - \theta_i \tag{3.29}$$

subject to:

$$Y_{ijm} = X_{ijm} \cdot \left(\sum_{l \in \mathcal{L}_{ij}} \gamma_{ijl} \cdot w_{ijlm} \right), \quad \forall j \in \mathcal{J}_i, \forall m \in \mathcal{M} \tag{3.30}$$

$$X_{ijm+1} = Y_{ijm} + U_m, \quad \forall j \in \mathcal{J}_i, \forall m \in \mathcal{M} \setminus \{M\} \tag{3.31}$$

$$\mathcal{R}^c_{ijm} = \frac{\mathcal{R}_{ij}(Y_{ij} + U_m)}{\mathcal{R}_{ij}(Y_{ij})}, \quad \forall j \in \mathcal{J}_i, \forall m \in \mathcal{M} \tag{3.32}$$

$$C_i = \sum_{j \in \mathcal{J}_i} \sum_{l \in \mathcal{L}_{ij}} \sum_{m \in \mathcal{M}} \left(c^v \cdot t_{ijl} \cdot w_{ijlm} + c^r_{ij} \cdot \int_{Y_{ijm}}^{Y_{ijm}+U_m} h_{ij}(t)\, dt \right) \tag{3.33}$$

$$\mathcal{T}_{im} = \sum_{j \in \mathcal{J}_i} \sum_{l \in \mathcal{L}_{ij}} t_{ijl} \cdot w_{ijlm}, \quad \forall m \in \mathcal{M} \tag{3.34}$$

$$\mathcal{R}^s_{im} \geq \mathcal{R}_{0m}, \quad \forall m \in \mathcal{M} \tag{3.35}$$

$$\mathcal{T}_{im} \leq \mathcal{T}_{0m}, \quad \forall m \in \mathcal{M} \tag{3.36}$$

$$\sum_{l \in \mathcal{L}_{ij}} w_{ijlm} = 1, \quad \forall j \in \mathcal{J}_i, \forall m \in \mathcal{M} \tag{3.37}$$

In the above optimization model, the objective function of Equation (3.29) represents the pricing operation. Constraints (3.30) allow to update the effective age of the components at the end of the maintenance break and constraints (3.31) update the components age at the end of each mission. Constraints (3.32) define the conditional reliability of component E_{ij} during mission m. Constraint (3.33) computes the total expected cost for subsystem i, while constraint (3.34) gives the time to perform the selected maintenance actions for subsystem i during break m. Constraints

(3.35) ensure that the reliability of subsystem i during mission m meets or exceeds the minimum reliability target. Constraints (3.36) require that the time to perform the selected maintenance actions for subsystem i during break m does not exceed the break duration. Constraints (3.37) ensure that a single maintenance action is selected for each component of subsystem i.

Solving the subproblems

The subproblems that must be solved to generate new columns are complex nonlinear and non-convex problems that are extremely difficult to solve to optimality, thus a heuristic was implemented to find approximate but good-quality solutions within a reasonable computation time. Specifically, an elitist genetic algorithm is used as it has been successfully applied to a wide range of combinatorial optimization problems (McCall, 2005; Talbi, 2009).

The genetic algorithm is inspired by the process of natural selection and begins with an initial population of size N_p individuals. Each individual has a fitness value and represents a potential solution to the optimization problem. The fitness value is determined by the fitness function which is defined by the subproblem objective function (3.29). An individual in the population is represented by a single solution matrix comprised of N_i rows and M columns. Each gene (element of solution matrix) takes a value from the set $\mathcal{L}_{ij}$ of possible maintenance levels and means that the j^{th} component of subsystem i is selected to receive maintenance level l, during the m^{th} scheduled break. The initial population is generated by assigning random values to elements of the solution matrix.

The elitism ranking procedure is used, where individuals in the population are ranked based on an increasing order of their fitness values. The best N_s solutions out of the N_p are selected to move on to the next generation. The ratio N_s/N_p is denoted by ξ_s and describes the number of elites in the population. The partial mapping crossover strategy is implemented and two blocks of the parents genomes are exchanged to produce two children. A set of N_c of new solutions resulting from the crossover operation is then injected into the next generation. The crossover rate is defined as $\xi_c = N_c/N_p$.

The mutation operation is performed by randomly selecting a parent from the previous population and then randomly modifying one of it's genes. The gene to be modified is selected at random, this process is repeated until N_m new solutions are formed. The mutation rate is defined as $\xi_m = N_m/N_p$. To increase the diversity of individuals in the population, additional solutions are randomly generated and injected into the next generation. The proportion of these extra solutions is denoted by ξ_e. All the mentioned ratios are selected such that their sum is equal to 100%. The genetic algorithm terminates when a given number of generations N_g is reached or when no solution improvement is achieved after a given number of generations N_{gi}.

3.6 Numerical experiments

In this section, four sets of numerical experiments are conducted to demonstrate the ability of the column generation approach to find valid and in many cases near optimal solutions. These experiments will also illustrate the benefit of using the proposed approach opposed to the BIP and other metaheuristics that have been suggested. The first set of experiments compares the column generation algorithm and the BIP approach with complete subsystem pattern information. The second experiment compares the column generation algorithm and another metaheuristic that has been suggested to solve the multimission SMP. Experiment 3 is an application to a real world coal transportation system and experiment 4 demonstrates the ability of the column generation algorithm to solve systems comprised of many components and subsystems in reasonable computation time. All experiments are run on a Intel™i5 2.9GHz desktop computer with 12GB of RAM running Windows 10™. All algorithms were coded in Python 3.8. The optimization runs were carried out by Gurobi 9.1 using gurobipy.

For all numerical experiments, five potential preventive maintenance levels will be considered: do-nothing ($l = 0$), imperfect maintenance ($0 < l < 4$) and replacement ($l = 4$). The age reduction coefficients for the different maintenance levels are reported in Table (3.1).

Table 3.1: Age reduction coefficients

l	0	1	2	3	4
γ_{ijl}	1.0	0.4	0.2	0.1	0.0

The default parameters used in the elitist genetic algorithm are set to $\xi_s = 0.1$, $\xi_c = 0.7$, $\xi_m = 0.1$ and $\xi_e = 0.1$ and the population $N_p = 200$. The algorithm terminates when no solution improvement is achieved after 150 generations or when the limit of 200 generations is reached

3.6.1 Set of experiments #1: comparison of the BIP and CG approach

This set of experiments considers a system comprised of three GA(1, 2) subsystems operating multiple missions. The shape and scale parameters, age at the start of the first maintenance break, time to perform the preventive maintenance levels, and the cost of minimal repair for each component are displayed in Table 3.2. The cost c_{ijl} induced by a maintenance action of level l when performed on a component E_{ij} is assumed to be proportional to the maintenance time t_{ijl}: $c_{ijl} = 15 \times t_{ijl}$. The system is required to operate multiple missions of equal duration. Table 3.4 displays the number of missions that the system must perform, the length of both the missions and breaks, as well as the minimum target reliability. Table 3.3 displays the system reliability for 5 missions when no maintenance actions are performed.

The CG and BIP approaches are both used to solve the multimission SMP for 10 different problem instances and the results are displayed in Table 3.4. The CG approach is run for multiple trials and the average and best costs C^* are reported, as well as the standard deviation (σ) of the expected costs, gap and computation time.

From the results displayed it is clear that the CG approach achieves valid and near optimal solutions for all problem instances as the largest gap is 2.4%. Although the solution time is higher for the CG algorithm for most problem instances, the BIP approach was unable to find a solution when $M = 5$ due to storage limitations (out-of memory error). From the results presented, one may observe that as the number of missions that the system is required to perform is increased the total cost and solution time also increases.

Table 3.5 displays the best selective maintenance plan for four instances of the problem. Looking at the maintenance actions selected during mission 1 for the first instance, actions [4, 4], [2, 3], [4, 0] imply that components 1 and 2 in subsystem 1 are replaced, components 1 and 2 in subsystem 2 receive IM levels $l = 2$ and $l = 3$ respectively, and component 1 in subsystem 3 is replaced.

Table 3.2: Component lifetime parameters and maintenance times

E_{ij}	β_{ij}	η_{ij}	X_{ij1}	t_{ijl}	t_{ij2}	t_{ij3}	t_{ij4}	c^r_{ij}
E_{11}	1.32	145	60	3.5	4	4.5	5	25
E_{12}	1.4	160	70	2	2.5	3	4	15
E_{21}	1.35	145	50	3	3.5	4	4.5	35
E_{22}	1.45	150	65	1.5	2	2.5	3.5	25
E_{31}	1.45	160	65	2	2.5	3	3.5	20
E_{32}	1.4	150	55	3.5	4	4.5	5	25

Table 3.3: Mission reliability when no maintenance actions are performed

m	1	2	3	4	5
$\mathcal{R}_m(\%)$	65.67	55.34	47.89	42.12	37.48

Table 3.4: Comparison of BIP and CG

				BIP		CG				
M	U_m	D_m	R_{0m}	Cost ($)	CPUt (s)	Av. cost ($)	Best cost ($)	σ($)	Gap (%)	CPUt (s)
2	60	30	0.80	639.6	0.1	653.2	647.8	3.6	1.3	58.3
2	60	20	0.75	433.1	0.1	444.5	440.5	3.3	1.7	39.9
2	60	20	0.65	216.5	0.1	216.5	216.5	0.0	0.0	15.7
3	60	20	0.60	327.6	3.9	336.6	335.6	1.1	2.4	43.3
3	60	30	0.80	957.6	3.6	983.9	979.5	4.5	2.3	148.2
3	60	20	0.75	664.9	4.3	673.8	670.5	3.9	0.8	101.7
4	60	30	0.80	1280.9	122.1	1308.6	1296.7	8.7	1.2	91.7
4	60	20	0.60	477.9	146.9	492.4	481.1	7.5	0.7	63.9
4	60	20	0.65	557.1	126.4	570.5	563.0	7.0	1.1	108.6
5	60	30	0.80	–	–	1654.4	1644.6	6.6	–	373.9

Table 3.5: Best SM plan

M	m	SM Plan	$\mathcal{R}_m(\%)$
2	1	[4, 4], [2, 3], [4, 0]	80.24
	2	[0, 3], [4, 3], [4, 4]	80.38
3	1	[0, 0], [0, 0], [0, 0]	65.67
	2	[0, 0], [0, 3], [0, 0]	60.24
	3	[0, 1], [0, 0], [4, 0]	61.68
4	1	[4, 4], [3, 2], [4, 0]	80.08
	2	[0, 3], [2, 4], [4, 4]	80.16
	3	[4, 3], [4, 3], [4, 0]	80.35
	4	[0, 3], [2, 4], [4, 4]	80.14
5	1	[0, 4], [4, 3], [4, 3]	80.68
	2	[4, 3], [0, 4], [2, 4]	80.09
	3	[0, 4], [4, 3], [4, 2]	80.39
	4	[4, 3], [0, 4], [2, 4]	80.09
	5	[0, 4], [4, 3], [4, 2]	80.39

3.6.2 Set of experiments #2: solving the multi-mission SMP for large parallel systems using CG

This set of experiments is used to compare the column generation approach and the genetic algorithm presented in Chaabane et al. (2020) in terms of solution quality. Three different subsystem structures are considered GA(1, 2), GA(1, 3) and GA(1, 5). For each trial table (3.7) displays the subsystem structure, the number of subsystems N, as well as the break and mission durations and minimum required reliability. For both the column generation and genetic algorithms the average and best cost is reported as well as the standard deviation (σ) and solution time. The number of columns generated for the column generation approach is also reported. The parameters used for the genetic algorithm presented by Chaabane et al. (2020) are displayed in Table 3.6.

The results show that the proposed CG approach achieves equal or better solutions than the genetic algorithm for 9 of the 12 experiments. For the GA(1, 2) subsystem structure the GA outperforms the CG approach in the first two runs and is able to find the optimal solution of $664.9 in run 1. However, as the system size increases the CG approach achieves better solutions than the GA in most cases. Although

Table 3.6: Parameters for genetic algorithm

Run	N_p	$\xi_s(\%)$	$\xi_c(\%)$	$\xi_m(\%)$	$\xi_e(\%)$	N_g	N_{gi}
1	100	0.1	0.7	0.1	0.1	2500	750
2	100	0.1	0.7	0.1	0.1	2500	750
3	100	0.1	0.7	0.1	0.1	2000	500
4	100	0.1	0.7	0.1	0.1	2000	500
5	300	0.1	0.7	0.1	0.1	5000	1500
6	300	0.1	0.7	0.1	0.1	5000	1500
7	300	0.1	0.7	0.1	0.1	5000	1500
8	300	0.1	0.7	0.1	0.1	5000	1500
9	100	0.1	0.7	0.1	0.1	2000	500
10	100	0.1	0.7	0.1	0.1	2000	500
11	300	0.1	0.7	0.1	0.1	5000	1500
12	300	0.1	0.7	0.1	0.1	5000	1500

the CG algorithm has longer solution time for larger systems, it is usually able to find better solutions. From these results, it appears that the CG algorithm is the preferable method for systems of moderate to large size.

Table 3.7: Comparison of GA and CG

						GA				CG				
Structure	N	M	U_m	D_m	$\mathcal{R}_{0m}$	Av. cost($)	Best cost($)	σ ($)	CPUt(s)	Av. cost($)	Best cost($)	σ ($)	Col.	CPUt(s)
GA(1, 2)	3	3	60	20	0.75	**671.0**	664.9	8.5	6.2	673.8	670.5	3.9	26	101.7
	3	3	60	30	0.80	**976.1**	970.3	7.8	206.8	983.9	979.5	4.5	26	148.2
	3	4	60	30	0.80	1318.5	1304.4	17.6	570.2	**1308.6**	1296.7	8.7	31	91.7
	3	5	60	30	0.80	1659.4	1648.7	13.5	7233.4	**1654.4**	1644.6	6.6	42	373.9
GA(1, 3)	2	5	100	15	0.90	413.0	411.7	3.1	58.6	**411.7**	411.7	0.0	15	72.1
	3	5	100	15	0.85	857.7	844.2	12.2	1352.8	**852.4**	849.9	3.3	26	126.0
	4	10	100	30	0.75	2895.7	2866.7	23.3	485.3	**2837.3**	2833.0	2.8	77	1273.2
	5	10	100	40	0.85	4398.7	4292.6	62.7	1808.5	**4193.1**	4177.8	14.9	157	1903.0
GA(1, 5)	2	5	150	15	0.90	**1106.7**	1103.4	4.5	982.1	1120.1	1110.5	16.8	24	356.7
	3	5	120	25	0.90	2943.8	2877.1	47.1	420.7	**2893.4**	2881.2	10.5	41	1095.4
	3	10	100	60	0.90	3705.3	3687.7	14.8	164.7	**3660.6**	3656.2	4.0	37	1804.3
	4	10	100	60	0.98	5477.2	5443.7	47.0	776.4	**5352.5**	5335.4	17.0	80	3484.9

3.6.3 Set of experiments #3: application to coal transportation system

In this set of experiments, the coal transportation system studied in Liu and Huang (2010) is considered. The coal transportation system reliability block diagram is shown in Figure 3.2. The system is comprised of 5 subsystems: 2 conveyors, 2 feeders and a stacker-reclaimer and is made up of 14 total components. The shape and scale parameters, age at the start of the first maintenance break, time to perform the preventive maintenance levels, and the cost of minimal repair for each component are displayed in Table 3.8. The repair crew variable cost c^v is set to \$15 and the system is required to perform 10 missions. Table 3.9 displays the system reliability for the 10 missions when no maintenance actions are performed.

Table 3.10 displays the average and best total expected costs, as well as the standard deviation (σ) and the number of maintenance actions performed when target reliability is varied. The number of columns generated through the CG procedure and solution time is also reported. As the target reliability is increased the total expected cost is increased, this is because more maintenance actions are performed to ensure the target reliability is met. Tables 3.11 and 3.12 display the detailed SM plans obtained for the cases where minimum target reliability is set to $\mathcal{R}_{0m} = 0.65$ and $\mathcal{R}_{0m} = 0.8$ respectively. As the target reliability is increased both the solution time and number of columns generated increases. Several experiments were conducted where mission length was varied from 70 time units to 100 time units and the other parameters fixed. The results from these experiments are shown in Table 3.13. As expected, when mission length is reduced fewer maintenance actions are performed and the total expected cost decreases.

3.6.4 Set of experiments #4: solving large systems comprised of k-out-of-n:G subsystems

The final set of experiments are used to demonstrate that the proposed column generation algorithm can be used to solve the multimission SMP for large systems comprised of both parallel and GA(k, n) subsystems. For all trials the system is required to perform 10 missions and the repair crew variable cost c^v is set to \$15. Table (3.14)

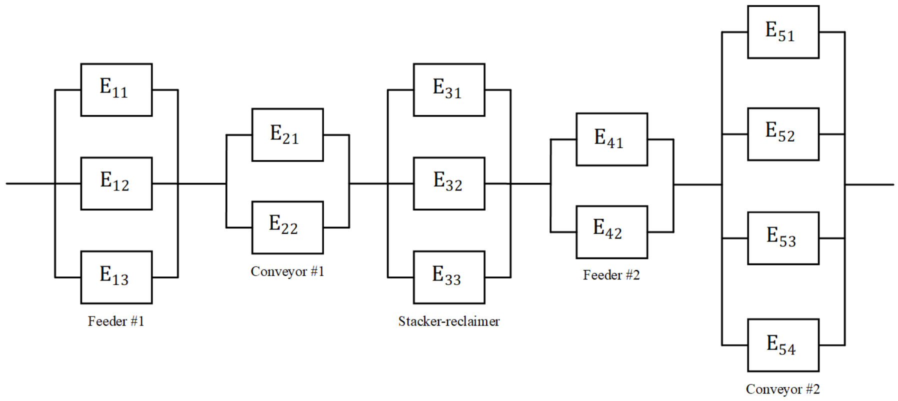

Figure 3.2: Coal transportation system RBD

Table 3.8: Component lifetime parameters and maintenance times

E_{ij}	β_{ij}	η_{ij}	X_{ij1}	t_{ijl}	t_{ij2}	t_{ij3}	t_{ij4}	c_{ij}^r
E_{11}	1.5	250	100	5.3	6	6.8	7.5	15
E_{12}	2.4	380	150	3	3.8	4.5	6	20
E_{13}	1.6	280	170	4.5	5.3	6	6.8	15
E_{21}	2.6	400	120	2.3	3	3.8	5.3	25
E_{22}	1.5	280	180	5.3	6	6.8	7.5	10
E_{31}	2.4	340	100	3	3.8	4.5	6	15
E_{32}	2.5	260	130	4.5	5.3	6	6.8	30
E_{33}	2.0	280	170	2.3	3	3.8	5.3	25
E_{41}	1.2	260	150	3	3.5	4	4.5	15
E_{42}	1.4	350	120	1.5	2	2.5	3.5	30
E_{51}	2.8	400	180	3	3.8	4.5	6	35
E_{52}	1.5	350	130	4.5	5.3	6	6.8	20
E_{53}	2.4	300	100	2.3	3	3.8	5.3	30
E_{54}	2.2	450	150	3	3.8	4.5	6	15

Table 3.9: Mission reliability when no maintenance actions are performed

m	1	2	3	4	5	6	7	8	9	10
$\mathcal{R}_m(\%)$	79.36	60.63	40.96	25.11	14.45	8.05	4.44	2.47	1.39	0.80

Table 3.10: Varying minimum required reliability

U_m	D_m	$\mathcal{R}_{0m}$	k^*	Av. cost (\$)	Best cost (\$)	σ (\$)	Columns	CPUt (s)
100	30	0.65	23	2869.1	2853.4	11.9	67	651.9
100	30	0.70	26	3008.7	2958.2	34.1	91	920.7
100	30	0.75	29	3185.5	3156.4	30.4	94	947.9
100	30	0.80	35	3509.2	3476.6	51.9	111	1355.0

k^*: total number of maintenance actions carried out

Table 3.11: Best SM plan obtained for $\mathcal{R}_{0m} = 0.65$

m	k^*_m	SM plan	$\mathcal{R}_m(\%)$
1	0	[0, 0, 0], [0, 0], [0, 0, 0], [0, 0], [0, 0, 0, 0]	79.36
2	1	[0, 0, 0], [0, 0], [0, 0, 2], [0, 0], [0, 0, 0, 0]	66.18
3	4	[0, 0, 0], [2, 0], [0, 4, 0], [0, 0], [3, 0, 2, 0]	70.93
4	2	[0, 2, 0], [0, 0], [2, 0, 0], [0, 0], [0, 0, 0, 0]	69.17
5	2	[0, 0, 0], [2, 0], [0, 0, 2], [0, 0], [0, 0, 0, 0]	65.80
6	4	[0, 0, 0], [0, 0], [0, 4, 0], [0, 4], [0, 0, 3, 2]	67.31
7	3	[0, 3, 0], [2, 0], [0, 0, 0], [0, 0], [3, 0, 0, 0]	72.18
8	2	[0, 0, 0], [0, 0], [3, 0, 2], [0, 0], [0, 0, 0, 0]	66.44
9	3	[0, 0, 0], [2, 0], [0, 4, 0], [0, 0], [0, 0, 2, 0]	65.59
10	2	[0, 3, 0], [1, 0], [0, 0, 0], [0, 0], [0, 0, 0, 0]	65.84

k^*_m: number of maintenance actions carried out during break m

Table 3.12: Best SM plan obtained for $\mathcal{R}_{0m} = 0.8$

m	k^*_m	SM plan	$\mathcal{R}_m(\%)$
1	1	[0, 0, 0], [0, 0], [0, 0, 3], [0, 0], [0, 0, 0, 0]	81.31
2	4	[0, 3, 0], [2, 0], [0, 4, 0], [0, 0], [2, 0, 0, 0]	83.71
3	4	[0, 0, 0], [3, 0], [3, 0, 2], [0, 0], [0, 0, 3, 0]	80.04
4	3	[0, 2, 0], [0, 0], [0, 3, 0], [0, 4], [0, 0, 0, 0]	81.25
5	4	[0, 0, 4], [2, 0], [0, 0, 2], [0, 0], [3, 0, 0, 0]	80.93
6	5	[0, 3, 0], [1, 0], [0, 3, 0], [4, 0], [0, 0, 2, 0]	83.30
7	3	[0, 0, 0], [2, 0], [4, 0, 0], [0, 0], [0, 0, 0, 3]	80.23
8	5	[0, 3, 0], [2, 0], [0, 0, 3], [0, 0], [2, 0, 2, 0]	80.39
9	3	[0, 0, 0], [2, 0], [0, 4, 0], [0, 4], [0, 0, 0, 0]	82.70
10	3	[0, 3, 0], [2, 0], [3, 0, 0], [0, 0], [0, 0, 0, 0]	80.30

k^*_m: number of maintenance actions carried out during break m

Table 3.13: Varying mission duration

U_m	D_m	$\mathcal{R}_{0m}$	k^*	Av. cost ($)	Best cost ($)	σ ($)	Col.	CPUt(s)
70	25	0.75	11	1779.9	1769.0	5.0	37	466.1
80	25	0.75	18	2197.5	2191.3	5.5	67	807.0
90	25	0.75	23	2670.9	2656.3	10.8	87	1265.2
100	25	0.75	30	3215.6	3161.3	48.9	98	1486.7

k^*: total number of maintenance actions carried out

displays the best cost, total number of columns generated, and computation time for all 18 problem instances. The results displayed highlight the ability of the proposed approach to provide valid maintenance decisions for large systems comprised of both parallel and k-out-of-n:G subsystems in a reasonable amount of time.

Table 3.14: Best cost for large systems comprised of both parallel and k-out-of-n:G subsystems

Structure	N	D_m	U_m	$\mathcal{R}_{0m}$	Best cost ($)	Col.	CPUt(s)
GA(1, 3)	6	40	100	0.75	4890.4	105	2271.4
	8	60	100	0.70	6862.5	132	3282.1
GA(1, 5)	6	40	100	0.80	5738.5	45	1619.7
	8	50	100	0.75	7850.3	78	3363.8
GA(2, 3)	6	100	100	0.60	9997.3	195	2691.4
	8	100	100	0.40	11419.5	168	1972.3
	10	150	100	0.20	12135.8	251	3845.5
GA(2, 5)	6	75	100	0.60	6483.1	107	2072.9
	8	100	100	0.60	9148.5	144	2756.7
	10	120	100	0.60	12291.4	159	2948.7
GA(1, 6)	6	100	120	0.90	8869.3	94	4057.3
	8	100	120	0.90	12123.3	154	6711.8
GA(2, 6)	6	80	100	0.65	7289.9	77	3687.5
	8	100	100	0.75	10533.9	151	7504.8
GA(1, 8)	6	150	100	0.85	14873.4	96	4438.3
	8	120	120	0.90	15416.6	83	5662.5
GA(4, 10)	6	200	100	0.80	17248.8	129	6476.2
	8	200	100	0.80	22649.7	203	11786.2

To further demonstrate the benefit of the column generation approach, the system comprised of the GA(1, 6) and GA(1, 8) subsystems is solved using the genetic algorithm proposed in Chaabane et al. (2020). Table 3.15 displays the comparison between the two approaches in terms of average and best cost obtained as well as the standard deviation (σ) and computation time. From the results displayed, the column generation approach achieves better solutions than the genetic algorithm for all problem instances. This comparison further confirms that the column generation approach is preferable for moderate and large scale systems.

The sets of experiments carried out above clearly show that the proposed column generation approach for the multimission SMP is efficient and yields valid maintenance decisions. The CG approach is superior in terms of solution quality compared to other heuristics that have been suggested. The CG algorithm was shown to be capable of providing maintenance decisions for systems comprised of many components and subsystems in a reasonable amount of time.

Table 3.15: Comparison of GA and CG

Structure	N	M	U_m	D_m	$\mathcal{R}_{0m}$	GA				CG				
						Avg cost ($)	Best cost ($)	σ ($)	CPUt(s)	Avg cost ($)	Best cost ($)	σ ($)	Col.	CPUt(s)
GA(1, 6)	6	10	120	100	0.90	9021.3	9000.4	26.2	1580.2	**8891.0**	8869.3	15.1	94	4057.3
	8	10	120	120	0.90	12498.0	12405.0	86.5	4128.7	**12155.0**	12123.3	26.0	154	6711.8
GA(1, 8)	6	10	150	100	0.85	15050.0	14998.0	36.3	6771.0	**14920.6**	14873.4	33.3	96	4438.3
	8	10	120	120	0.90	15673.3	15628.0	40.9	4695.8	**15467.6**	15416.6	34.5	83	5662.5

3.7 Conclusion

This chapter introduced a new solution method for the multimission SMP in the form of a hybrid column generation and genetic algorithm. The genetic algorithm is used to solve the complex column generating subproblems and provides high quality solutions quickly. The proposed solution method was shown to obtain equal or better solutions than the genetic algorithm proposed in Chaabane et al. (2020) for the majority of problem instances that were solved. The column generation algorithm was applied to the real world application of determining the SM plan for a coal transportation system required to operate consecutive missions. Multiple numerical experiments were run to show that the CG approach can be used to solve large systems comprised of parallel and k-out-of-n:G subsystems.

There are several areas and opportunities where the work presented in this chapter could be extended and improved. The proposed column generation algorithm will not guarantee optimality because the subproblems are solved using a heuristic rather than an exact solution technique. Developing an exact method to solve the very difficult subproblems that is also efficient would be of great interest. If an exact solution method could be developed such as dynamic programming or other decomposition methods, then a branch-and-price framework could be applied to find the global optimal solution. Other work on the research presented includes applying different evolutionary algorithms, swarm algorithms and metaheuristics to solve the subproblems, as well as using parallel algorithms to reduce computation time.

Chapter 4

Conclusions and research perspectives

The contributions of this Master's thesis are in the area of selective maintenance for binary state systems. Selective maintenance has the potential to significantly help many industries with large maintenance operations like aerospace, mining, transportation, manufacturing, petro-chemical plants, and offshore energy production systems. The contributions of this dissertation to the literature on SMP are made in two separate but dependent research themes. These themes are presented and fully discussed in Chapters 2 and 3. Chapter 2 introduced a new framework for modeling and optimizing the joint selective maintenance and orienteering problem. Chapter 3 proposed and developed a new column generation approach to solve the multimission selective maintenance problem. The conclusions and future research extensions for the two themes are discussed below.

The objective of the first theme was to introduce and optimally solve the joint selective maintenance and orienteering problem. The JSMOP filled a very important gap which was that all SM models assumed that the maintenance decisions are made and performed on systems which are all at the same location and have access to ample repair crews. This assumption is not always valid as many industrial systems can be geographically dispersed and their maintenance may then require that repair crews travel under transportation and cost constraints. The proposed joint selective maintenance and orienteering framework simultaneously makes the decisions of what systems to visit, the components to maintain, the maintenance levels to be performed, the assignment of repair crews, and their routing with the goal of minimizing total cost while satisfying a minimum required reliability threshold. To solve the resulting difficult optimization problem the two-phase solution approach presented in Diallo et al. (2018) was successfully used to identify the optimal SM and routing decisions. Multiple numerical experiments were conducted to demonstrate that the proposed joint approach can achieve better solutions than the traditional approach of first selecting the maintenance actions to be performed followed by the routing decisions (sequential approach).

Future extensions of the JSMOP could include using mean availability as the desired performance indicator. Mean availability refers to the percentage of time that the system is in the functioning state during the mission and is a very common metric that is used in practice. Developing a model where the systems must meet or exceed a mean availability would be an innovative extension of the SMP. Another area of future work could be developing methods to solve very large instances of the JSMOP. Decomposition methods such as column generation and Lagrangian relaxation have been utilized to solve many large-scale optimization problems and could potentially be applied to the JSMOP.

The second theme discussed in Chapter 3 presented a column generation approach for the multimission SMP. As stated by Schneider (2006) regarding the selective maintenance problem, the solution procedures and heuristics that allow for larger, more realistic-sized problems should be further considered. Column generation has been one of the biggest success stories in large-scale integer programming and has been applied to a wide range of different problems. The column generation approach presented in theme 2 is based on the classical framework where a restricted master problem is solved given only a subset of feasible solutions together with multiple column generating subproblems that are used to find new potential solutions. Due to the complexity of the subproblems, an elitist genetic algorithm was used to find high quality solutions quickly. Multiple numerical experiments were also performed to demonstrate the ability of the proposed approach to obtain near optimal solutions and find valid maintenance decisions for systems of very large size in reasonable computation time. The numerical experiments also showed that the column generation approach outperformed other heuristics for the majority of the problem instances that were solved.

Future work dealing with theme 2 would be developing an exact method to solve the complex subproblems. Because the subproblems are solved using an evolutionary algorithm rather than an exact method, optimality cannot be guaranteed. If an exact and efficient solution method exists then a branch-and-price framework could be applied to find the optimal solution. It would also be of interest to apply different evolutionary algorithms, swarm algorithms and metaheuristics to solve the subproblems. It would also be very useful to apply the large-scale optimization technique of column generation to other variants of the selective maintenance problem like fleet-level selective maintenance.

There has been significant research on the selective maintenance problem under uncertainty. Liu et al. (2018) study the SMP when the break and maintenance durations are stochastic rather than deterministic. Another interesting extension of the SMP would be to develop new robust and distributionally robust models and their related solution techniques such as decomposition and relaxation schemes to solve large-scale SMP under uncertainty.

Fleet level selective maintenance is an extension of the classic SMP where there is opportunity for further work. All fleet level models assume that the break and mission durations for all systems in the fleet are identical. In many practical applications this may not be valid. An important extension would be to consider a situation in which the break and mission durations are not aligned for each system in the fleet. Finally, all papers that deal with the joint selective maintenance and repairperson assignment problem ignore workload balancing. In many scenarios it would be of importance that all repairpersons/crews are given a relatively even distribution of work.

References

Afsar, H. M. and Labadie, N. (2013). Team orienteering problem with decreasing profits. *Electronic Notes in Discrete Mathematics*, 41:285–293.

Al-Yakoob, S. M. and Sherali, H. D. (2008). A column generation approach for an employee scheduling problem with multiple shifts and work locations. *Journal of the Operational Research Society*, 59(1):34–43.

Archetti, C., Bianchessi, N., and Speranza, M. G. (2013). The capacitated team orienteering problem with incomplete service. *Optimization letters*, 7(7):1405–1417.

Archetti, C., Feillet, D., Hertz, A., and Speranza, M. G. (2009). The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(6):831–842.

Arulmozhi, G. (2002). Exact equation and an algorithm for reliability evaluation of k-out-of-n: G system. *Reliability Engineering & System Safety*, 78(2):87–91.

Campbell, A. M., Gendreau, M., and Thomas, B. W. (2011). The orienteering problem with stochastic travel and service times. *Annals of Operations Research*, 186(1):61–81.

Cao, W., Jia, X., Hu, Q., Song, W., and Ge, H. (2017). Selective maintenance for maximising system availability: a simulation approach. *International Journal of Innovative Computing and Applications*, 8(1):12–20.

Cao, W., Jia, X., Hu, Q., Zhao, J., and Wu, Y. (2018). A literature review on selective maintenance for multi-unit systems. *Quality and Reliability Engineering International*, pages 1–22.

Cao, W., Song, W., Hu, Q., and Du, Y. (2016). An exact method for solving selective maintenance problems considering imperfect maintenance. In *Intelligent Networking and Collaborative Systems (INCoS), 2016 International Conference on*, pages 522–526. IEEE.

Carroll, J., McDonald, A., and McMillan, D. (2016). Failure rate, repair time and unscheduled O&M cost analysis of offshore wind turbines. *Wind Energy*, 19(6):1107–1119.

Cassady, C. R., Murdock Jr, W. P., and Pohl, E. A. (2001a). Selective maintenance for support equipment involving multiple maintenance actions. *European Journal of Operational Research*, 129(2):252–258.

Cassady, C. R., Pohl, E. A., and Murdock, W. P. (2001b). Selective maintenance modeling for industrial systems. *Journal of Quality in Maintenance Engineering.*

Chaabane, K., Khatab, A., Diallo, C., Aghezzaf, E.-H., and Venkatadri, U. (2020). Integrated imperfect multimission selective maintenance and repairpersons assignment problem. *Reliability Engineering & System Safety*, 199:106895.

Chao, I.-M., Golden, B. L., and Wasil, E. A. (1996). A fast and effective heuristic for the orienteering problem. *European journal of operational research*, 88(3):475–489.

Dai, L., Stålhane, M., and Utne, I. B. (2015). Routing and scheduling of maintenance fleet for offshore wind farms. *Wind Engineering*, 39(1):15–30.

Dao, C. D. and Zuo, M. J. (2016). Selective maintenance for multi-state series systems with s-dependent components. *IEEE Transactions on Reliability*, 65(2):525–539.

Dao, C. D. and Zuo, M. J. (2017). Optimal selective maintenance for multi-state systems in variable loading conditions. *Reliability engineering & system safety*, 166:171–180.

Diallo, C., Venkatadri, U., Khatab, A., and Liu, Z. (2018). Optimal selective maintenance decisions for large serial k-out-of-n: G systems under imperfect maintenance. *Reliability Engineering & System Safety*, 175:234–245.

Diallo, C., Venkatadri, U., Khatab, A., Liu, Z., and Aghezzaf, E.-H. (2019). Optimal joint selective imperfect maintenance and multiple repairpersons assignment strategy for complex multicomponent systems. *International Journal of Production Research*, 57(13):4098–4117.

Dunbar, M., Belieres, S., Shukla, N., Amirghasemi, M., Perez, P., and Mishra, N. (2020). A genetic column generation algorithm for sustainable spare part delivery: application to the sydney droppoint network. *Annals of Operations Research*, 290(1):923–941.

Elsayed, E. A. (2012). *Reliability Engineering.* Wiley Publishing, 2nd edition.

Faiz, T. I., Vogiatzis, C., and Noor-E-Alam, M. (2019). A column generation algorithm for vehicle scheduling and routing problems. *Computers & Industrial Engineering*, 130:222–236.

Fomin, F. V. and Lingas, A. (2002). Approximation algorithms for time-dependent orienteering. *Information Processing Letters*, 83(2):57–62.

Gilmore, P. C. and Gomory, R. E. (1961). A linear programming approach to the cutting stock problem. *Operations research*, 9(6):849–859.

Gilmore, P. C. and Gomory, R. E. (1963). A linear programming approach to the cutting stock problem-part II. *Operations research*, 11(6):863–888.

Goel, A. and Meisel, F. (2013). Workforce routing and scheduling for electricity network maintenance with downtime minimization. *European Journal of Operational Research*, 231(1):210–228.

Golden, B. L., Levy, L., and Vohra, R. (1987). The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318.

IEA (2020). Offshore wind. `https://www.iea.org/reports/offshore-wind`.

Ikonen, T. J., Mostafaei, H., Ye, Y., Bernal, D. E., Grossmann, I. E., and Harjunkoski, I. (2020). Large-scale selective maintenance optimization using bathtub-shaped failure rates. *Computers & Chemical Engineering*, 139:106876.

Ilhan, T., Iravani, S. M., and Daskin, M. S. (2008). The orienteering problem with stochastic profits. *Iie Transactions*, 40(4):406–421.

Irawan, C. A., Eskandarpour, M., Ouelhadj, D., and Jones, D. (2021). Simulation-based optimisation for stochastic maintenance routing in an offshore wind farm. *European Journal of Operational Research*, 289(3):912–926.

Ivanov, D., Dolgui, A., Sokolov, B., Werner, F., and Ivanova, M. (2016). A dynamic model and an algorithm for short-term supply chain scheduling in the smart factory industry 4.0. *International Journal of Production Research*, 54(2):386–402.

Jia, C. and Zhang, C. (2020). Joint optimization of maintenance planning and workforce routing for a geographically distributed networked infrastructure. *IISE Transactions*, 52(7):732–750.

Jiang, T. and Liu, Y. (2020a). Robust selective maintenance strategy under imperfect observations: A multi-objective perspective. *IISE Transactions*, 52(7):751–768.

Jiang, T. and Liu, Y. (2020b). Selective maintenance strategy for systems executing multiple consecutive missions with uncertainty. *Reliability Engineering & System Safety*, 193:106632.

Kamal, M., Modibbo, U. M., AlArjani, A., and Ali, I. (2021). Neutrosophic fuzzy goal programming approach in selective maintenance allocation of system reliability. *Complex & Intelligent Systems*, 7(2):1045–1059.

Kaynak, O. (2007). The exhilarating journey from industrial electronics to industrial informatics. In *2007 2nd IEEE Conference on Industrial Electronics and Applications*, pages xli–xlii. IEEE.

Khatab, A., Aghezzaf, E. H., Diallo, C., and Djelloul, I. (2017). Selective maintenance optimisation for series-parallel systems alternating missions and scheduled breaks with stochastic durations. *International Journal of Production Research*, 55(10):3008–3024.

Khatab, A. and Ait-Kadi, D. (2008). Selective maintenance policy for multi-mission multi-state series-parallel systems. In 4^{th} *International Conference on Advances in Mechanical Engineering and Mechanics (ICAMEM2008), Tunisia*.

Khatab, A., Diallo, C., Aghezzaf, E.-H., and Venkatadri, U. (2018a). Condition-based selective maintenance for stochastically degrading multi-component systems under periodic inspection and imperfect maintenance. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 232(4):447–463.

Khatab, A., Diallo, C., Aghezzaf, E.-H., and Venkatadri, U. (2020). Optimization of the integrated fleet-level imperfect selective maintenance and repairpersons assignment problem. *Journal of Intelligent Manufacturing*, pages 1–16.

Khatab, A., Diallo, C., Venkatadri, U., Liu, Z., and Aghezzaf, E.-H. (2018b). Optimization of the joint selective maintenance and repairperson assignment problem under imperfect maintenance. *Computers & Industrial Engineering*, 125:413–422.

Kovacs, A. A., Parragh, S. N., Doerner, K. F., and Hartl, R. F. (2012). Adaptive large neighborhood search for service technician routing and scheduling problems. *Journal of scheduling*, 15(5):579–600.

Kuo, W. and Zuo, M. J. (2003). *Optimal reliability modeling: principles and applications*. John Wiley & Sons.

Kusiak, A. (2018). Smart manufacturing. *International Journal of Production Research*, 56(1-2):508–517.

Lin, D., Zuo, M. J., and Yam, R. C. M. (2000). General sequential imperfect preventive maintenance models. *International Journal of Reliability, Quality and Safety Engineering*, 7(3):253–266.

Liu, M., Haghani, A., and Toobaie, S. (2010). Genetic algorithm–based column generation approach to passenger rail crew scheduling. *Transportation research record*, 2159(1):36–43.

Liu, Y., Chen, Y., and Jiang, T. (2018). On sequence planning for selective maintenance of multi-state systems under stochastic maintenance durations. *European Journal of Operational Research*, 268(1):113–127.

Liu, Y. and Huang, H.-Z. (2010). Optimal selective maintenance strategy for multi-state systems under imperfect maintenance. *IEEE Transactions on Reliability*, 59(2):356–367.

López-Santana, E., Akhavan-Tabatabaei, R., Dieulle, L., Labadie, N., and Medaglia, A. L. (2016). On the combined maintenance and routing optimization problem. *Reliability Engineering & System Safety*, 145:199–214.

Lübbecke, M. E. and Desrosiers, J. (2005). Selected topics in column generation. *Operations research*, 53(6):1007–1023.

Lust, T., Roux, O., and Riane, F. (2009). Exact and heuristic methods for the selective maintenance problem. *European Journal of Operational Research*, 197:1166–1177.

Maillart, L. M., Cassady, C. R., Rainwater, C., and Schneider, K. (2009). Selective maintenance decision-making over extended planning horizons. *IEEE Transactions on Reliability*, 58(3):462–469.

Malik, M. (1979). Reliable preventive maintenance scheduling. *AIIE transactions*, 11(3):221–228.

Mathlouthi, I., Gendreau, M., and Potvin, J.-Y. (2018). Mixed integer linear programming for a multi-attribute technician routing and scheduling problem. *INFOR: Information Systems and Operational Research*, 56(1):33–49.

McCall, J. (2005). Genetic algorithms for modelling and optimisation. *Journal of computational and Applied Mathematics*, 184(1):205–222.

Nakagawa, T. (1988). Sequential imperfect preventive maintenance policies. *IEEE Transactions on Reliability*, 37(3):295–308.

Pandey, M., Zuo, M. J., and Moghaddass, R. (2013a). Selective maintenance modeling for a multistate system with multistate components under imperfect maintenance. *Iie Transactions*, 45(11):1221–1234.

Pandey, M., Zuo, M. J., and Moghaddass, R. (2016). Selective maintenance scheduling over a finite planning horizon. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 230(2):162–177.

Pandey, M., Zuo, M. J., Moghaddass, R., and Tiwari, M. K. (2013b). Selective maintenance for binary systems under imperfect repair. *Reliability Engineering and System Safety*, 113:42–51.

Range, T. M., Kozlowski, D., and Petersen, N. C. (2019). Dynamic job assignment: A column generation approach with an application to surgery allocation. *European Journal of Operational Research*, 272(1):78–93.

Rice, W., Cassady, C., and Nachlas, J. (1998). Optimal maintenance plans under limited maintenance time. In *Proceedings of the seventh industrial engineering research conference*, pages 1–3.

Rice, W. F. (1999). *Optimal selective maintenance decisions for series systems*. PhD thesis, Mississippi State University. Department of Industrial Engineering.

Röckmann, C., Lagerveld, S., and Stavenuiter, J. (2017). Operation and maintenance costs of offshore wind farms and potential multi-use platforms in the dutch north sea. In *Aquaculture Perspective of Multi-Use Sites in the Open Ocean*, pages 97–113. Springer, Cham.

Schneider, K. and Cassady, C. R. (2015). Evaluation and comparison of alternative fleet-level selective maintenance models. *Reliability Engineering & System Safety*, 134:178–187.

Schneider, K. R. (2006). *Comparison of Alternative Fleet-level Selective Maintenance Models*. ProQuest.

Shahraki, A., Yadav, O., and Vogiatzis, C. (2020). Selective maintenance optimization for multi-state systems considering stochastically dependent components and stochastic imperfect maintenance actions. *Reliability Engineering & System Safety*, 196:106738.

Sharma, P., Kulkarni, M. S., and Yadav, V. (2017). A simulation based optimization approach for spare parts forecasting and selective maintenance. *Reliability Engineering & System Safety*.

Stålhane, M., Hvattum, L. M., and Skaar, V. (2015). Optimization of routing and scheduling of vessels to perform maintenance at offshore wind farms. *Energy Procedia*, 80:92–99.

Talbi, E.-G. (2009). *Metaheuristics: from design to implementation*. Wiley.

Tang, H., Miller-Hooks, E., and Tomastik, R. (2007). Scheduling technicians for planned maintenance of geographically distributed equipment. *Transportation Research Part E: Logistics and Transportation Review*, 43(5):591–609.

Tsiligirides, T. (1984). Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35(9):797–809.

Vansteenwegen, P., Souffriau, W., and Van Oudheusden, D. (2011). The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1–10.

Verbeeck, C., Vansteenwegen, P., and Aghezzaf, E.-H. (2017). The time-dependent orienteering problem with time windows: a fast ant colony system. *Annals of Operations Research*, 254(1):481–505.

Xu, L. D., Xu, E. L., and Li, L. (2018). Industry 4.0: state of the art and future trends. *International Journal of Production Research*, 56(8):2941–2962.

Xu, Q.-z., Guo, L.-m., Shi, H.-p., and Wang, N. (2016). Selective maintenance problem for series–parallel system under economic dependence. *Defence technology*, 12(5):388–400.

Xu, Q.-Z., Guo, L.-M., Wang, N., and Fei, R. (2015). Recent advances in selective maintenance from 1998 to 2014. *Journal of Donghua University (English Edition)*, 32(6):986–994.

Zhang, S., Ohlmann, J. W., and Thomas, B. W. (2014). A priori orienteering with time windows and stochastic wait times at customers. *European Journal of Operational Research*, 239(1):70–79.

Zhang, X., Chen, J., Han, B., and Li, J. (2019). multi-mission selective maintenance modelling for multistate systems over finite time horizon. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 233(6):1040–1059.

Zhao, J. and Zeng, J.-C. (2016). Maintenance strategy for stochastic selective maintenance of a two-state system. *International Journal of Wireless and Mobile Computing*, 11(4):302–308.

Zhu, H., Liu, F., Shao, X., Liu, Q., and Deng, Y. (2011). A cost-based selective maintenance decision-making method for machining line. *Quality and Reliability Engineering International*, 27:191–201.

Zia, L. and Coit, D. W. (2010). Redundancy allocation for series-parallel systems using a column generation approach. *IEEE Transactions on Reliability*, 59(4):706–717.

Appendix A

Component information for Chapter 2 experiments

E_{ijk}	u_{ijk}	B_{ijk}	η_{ijk}	β_{ijk}	t^c_{ijk1}	t^c_{ijk2}	t^c_{ijk3}	t^p_{ijk2}	t^p_{ijk3}	c^c_{ijk1}	c^c_{ijk2}	c^c_{ijk3}	c^p_{ijk2}	c^p_{ijk3}
E_{111}	1	10	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{112}	0	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{113}	1	6	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{114}	0	10	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{115}	1	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{121}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{122}	0	4	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{123}	1	12	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{124}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{131}	1	12	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{132}	0	16	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{133}	1	14	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{141}	1	10	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{142}	1	5	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{211}	1	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{212}	0	6	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{213}	1	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{214}	1	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{215}	1	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{221}	1	8	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{222}	1	18	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{223}	0	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{224}	1	9	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{231}	1	7	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{232}	1	6	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{233}	1	12	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{241}	1	14	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{242}	1	12	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{311}	1	11	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200

Continued on Next Page

E_{ijk}	u_{ijk}	B_{ijk}	η_{ijk}	β_{ijk}	t^c_{ijk1}	t^c_{ijk2}	t^c_{ijk3}	t^p_{ijk2}	t^p_{ijk3}	c^c_{ijk1}	c^c_{ijk2}	c^c_{ijk3}	c^p_{ijk2}	c^p_{ijk3}
E_{312}	0	16	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{313}	1	7	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{314}	1	10	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{315}	0	16	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{321}	1	12	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{322}	0	16	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{323}	1	12	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{324}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{331}	1	12	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{332}	0	16	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{333}	1	14	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{341}	0	10	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{342}	0	12	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{411}	1	6	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{412}	0	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{413}	0	6	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{414}	1	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{415}	1	4	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{421}	1	8	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{422}	0	16	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{423}	1	6	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{424}	0	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{431}	1	9	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{432}	1	7	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{433}	1	10	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{441}	1	8	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{442}	1	6	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{511}	1	10	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{512}	0	5	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{513}	1	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{514}	1	9	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{515}	1	11	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{521}	1	12	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{522}	0	9	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{523}	1	12	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225

Continued on Next Page

E_{ijk}	u_{ijk}	B_{ijk}	η_{ijk}	β_{ijk}	t^c_{ijk1}	t^c_{ijk2}	t^c_{ijk3}	t^p_{ijk2}	t^p_{ijk3}	c^c_{ijk1}	c^c_{ijk2}	c^c_{ijk3}	c^p_{ijk2}	c^p_{ijk3}
E_{524}	0	8	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{531}	1	18	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{532}	1	8	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{533}	1	14	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{541}	0	8	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{542}	0	8	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{611}	1	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{612}	0	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{613}	1	4	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{614}	1	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{615}	0	12	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{621}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{622}	1	8	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{623}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{624}	1	20	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{631}	1	10	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{632}	0	8	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{633}	1	16	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{641}	1	12	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{642}	1	8	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{711}	0	7	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{712}	0	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{713}	0	5	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{714}	1	8	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{715}	1	14	26	2	0.75	2	3.5	2	3.5	50	100	200	100	200
E_{721}	1	15	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{722}	0	16	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{723}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{724}	1	10	28	3.5	1	2.5	4	2.5	4	75	125	225	125	225
E_{731}	1	10	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{732}	0	9	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{733}	1	13	26	4	0.75	1.75	3.75	1.75	3.75	80	115	220	115	220
E_{741}	0	14	25	3	1	2.25	4	2.25	4	50	100	225	100	225
E_{742}	0	15	25	3	1	2.25	4	2.25	4	50	100	225	100	225

Appendix B

Details of the optimal maintenance actions obtained for Chapter 2 experiment set #5

Table B.1: Maintenance actions performed for $\mathcal{R}_0 = 86\%$: case of Experiments #5

OWT i	Patterns selected	$\mathcal{R}^*(\%)$
1	(0, 0, 0, 1, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	88.61
2	(0, 1, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	87.81
3	(0, 1, 0, 0, 1), (0, 0, 0, 0), (0, 0, 0), (1, 1)	86.54
4	(0, 0, 1, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	86.32
5	(0, 0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0), (1, 1)	91.11
6	(0, 1, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	90.88
7	(1, 0, 1, 0, 0), (0, 0, 0, 0), (0, 0, 0), (1, 2)	87.37

Table B.2: Maintenance actions performed for $\mathcal{R}_0 = 90\%$: case of Experiments #5

OWT i	Patterns selected	$\mathcal{R}^*(\%)$
1	(0, 1, 0, 1, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	90.44
2	(0, 1, 0, 0, 0), (0, 0, 1, 0), (0, 0, 0), (0, 0)	92.01
3	(0, 0, 0, 0, 1), (0, 1, 0, 0), (0, 1, 0), (1, 1)	90.18
4	(0, 0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0), (0, 0)	91.55
5	(0, 0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 0), (1, 1)	91.11
6	(0, 1, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	90.88
7	(1, 1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 0), (2, 1)	90.55

Table B.3: Maintenance actions performed for $\mathcal{R}_0 = 94\%$: case of Experiments #5

OWT i	Patterns selected	$\mathcal{R}^*(\%)$
1	(0, 1, 0, 0, 0), (0, 1, 0, 0), (0, 1, 0), (0, 0)	94.93
2	(0, 1, 0, 0, 0), (0, 0, 1, 0), (0, 0, 0), (2, 0)	95.96
3	(0, 1, 0, 0, 1), (0, 1, 0, 0), (0, 1, 0), (1, 2)	94.69
4	(0, 1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 0), (0, 0)	94.38
5	(0, 0, 0, 0, 0), (0, 1, 0, 1), (0, 0, 0), (1, 1)	94.49
6	(0, 1, 0, 0, 1), (0, 0, 0, 0), (0, 1, 0), (0, 0)	95.24
7	(1, 1, 1, 0, 0), (0, 1, 0, 0), (0, 0, 0), (3, 0)	94.45

Table B.4: Maintenance actions performed for $\mathcal{R}_0 = 98\%$: case of Experiments #5

OWT i	Patterns selected	$\mathcal{R}^*(\%)$
1	(0, 1, 0, 1, 0), (0, 1, 0, 0), (0, 2, 0), (2, 0)	98.32
2	(0, 1, 0, 0, 0), (0, 0, 1, 0), (0, 0, 0), (0, 3)	98.12
3	(0, 1, 0, 0, 1), (0, 2, 0, 2), (0, 2, 0), (1, 3)	98.06
4	(0, 1, 1, 0, 0), (0, 1, 0, 1), (0, 0, 0), (0, 0)	98.07
5	(0, 1, 0, 0, 0), (0, 1, 0, 1), (0, 0, 0), (2, 2)	98.02
6	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	85.17
7	(1, 1, 1, 0, 0), (0, 1, 0, 2), (0, 1, 0), (3, 1)	98.02

Table B.5: Maintenance actions performed for $\mathcal{R}_0 = 99\%$: case of Experiments #5

OWT i	Patterns selected	$\mathcal{R}^*(\%)$
1	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	81.98
2	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	85.34
3	(0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0)	0
4	(0, 1, 1, 0, 0), (0, 2, 0, 1), (0, 0, 0), (2, 0)]	99.03
5	(0, 1, 0, 0, 0), (0, 2, 0, 1), (2, 0, 0), (2, 3)	99.00
6	(0, 1, 0, 0, 2), (0, 0, 0, 2), (0, 2, 0), (3, 0)	99.05
7	(1, 1, 1, 0, 0), (0, 2, 0, 2), (0, 2, 0), (2, 3)	99.03