SIMULATION AND OPTIMIZATION OF WIND FARM OPERATIONS UNDER STOCHASTIC CONDITIONS

A Dissertation

by

EUNSHIN BYON

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2010

Major Subject: Industrial Engineering

Simulation and Optimization of Wind Farm Operations under Stochastic Conditions

SIMULATION AND OPTIMIZATION OF WIND FARM OPERATIONS UNDER STOCHASTIC CONDITIONS

A Dissertation

by

EUNSHIN BYON

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Yu Ding
Committee Members,	Natarajan Gautam
	Lewis Ntaimo
	Chanan Singh
Head of Department,	Brett A. Peters

May 2010

Major Subject: Industrial Engineering

ABSTRACT

Simulation and Optimization of Wind Farm Operations under Stochastic Conditions. (May 2010)

Eunshin Byon, B. S., Korean Advanced Science and Technology;

M.S., Korean Advanced Science and Technology

Chair of Advisory Committee: Dr. Yu Ding

This dissertation develops a new methodology and associated solution tools to achieve optimal operations and maintenance strategies for wind turbines, helping reduce operational costs and enhance the marketability of wind generation. The integrated framework proposed includes two optimization models for enabling decision support capability, and one discrete event-based simulation model that characterizes the dynamic operations of wind power systems. The problems in the optimization models are formulated as a partially observed Markov decision process to determine an optimal action based on a wind turbine's health status and the stochastic weather conditions.

The first optimization model uses homogeneous parameters with an assumption of stationary weather characteristics over the decision horizon. We derive a set of closed-form expressions for the optimal policy and explore the policy's monotonicity. The second model allows time-varying weather conditions and other practical aspects. Consequently, the resulting strategy are season-dependent. The model is solved using a backward dynamic programming method. The benefits of the optimal policy are highlighted via a case study that is based upon field data from the literature and industry. We find that the optimal policy provides options for cost-effective actions, because it can be adapted to a variety of operating conditions.

Our discrete event-based simulation model incorporates critical components, such as a wind turbine degradation model, power generation model, wind speed model, and maintenance model. We provide practical insights gained by examining different maintenance strategies. To the best of our knowledge, our simulation model is the first discrete-event simulation model for wind farm operations.

Last, we present the integration framework, which incorporates the optimization results in the simulation model. Preliminary results reveal that the integrated model has the potential to provide practical guidelines that can reduce the operation costs as well as enhance the marketability of wind energy.

To my family

ACKNOWLEDGMENTS

I am heartily thankful to my Ph.D. advisor, Dr. Yu Ding. His inspirational guide and support from the preliminary to the concluding level enabled me to develop an understanding of the subject. As a result, my research life became smooth and rewarding for me. I would also like to acknowledge my committee members: Dr. Natarajan Gautam, Dr. Lewis Ntaimo and Dr. Chanan Singh for their constant encouragement. A special note of thanks goes to Dr. Lewis Ntaimo for his helpful discussions at various times.

In my daily work I have been blessed with a challenging but friendly group of fellow students. My lab mates, Jung-Jin Cho, Haifeng (Heidi) Xia, Yuan Ren, Abhishek K. Shrivastava, Chiwoo Park and Arash Pourhabib, provided insight that guided and challenged my thinking, substantially improving my research product. I am also indebted to many of my student colleagues. My thanks go to Youngmyung Ko, Chaehwa Lee, Soondo Hong and Moya Hiram and many more, with whom I could share many intellectual thoughts. Additionally, the Department of Industrial and Systems Engineering provides an stimulating environment in which to learn and grow. I am especially grateful to Judy Meeks for always being so kind in assisting me in many different ways.

My deepest appreciation goes to my family. My sincere thanks goes to my husband Phil for his endless love and patience. I remember his constant support when I encountered difficulties. My beautiful kids, David and Danielle, have been a strong force during my doctoral studies. I thank my mother and my in-laws for their unconditional support, belief and affection.

Above all, thank you, God. May your name be exalted, honored, and glorified.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

I.1. Motivation

Propelled by the need to mitigate climate change and high energy costs, wind power has become one of the fastest growing renewable energy sources around the world. Worldwide, wind energy increased from 18 GW to 152 GW over the past decade. In the US, total capacity of wind energy rose 45% in 2007 and is forecasted to nearly triple by 2012 (American Wind Energy Association, 2008). According to NERC (2009), approximately 260 GW of new renewable nameplate capacity is projected in the US during 2009-2018. Roughly 96% of this total is estimated to be wind energy. In fact, NERC projects that wind power alone will account for 18% of the US total resource mix by 2018.

However, despite the vast capacity of global wind power reserve, the share of wind energy comprises only a small portion of the current energy market. A key factor for enhancing the marketability of wind energy is to reduce operations and maintenance (O&M) costs (Vachon, 2002, Walford, 2006, Wiser and Bolinger, 2008). According to Walford (2006), the contribution of O&M costs to the total energy production cost could be as much as 20% for a wind farm. Vachon (2002) shows that O&M costs can account for 75-90% of investment costs based on a 20-year life cycle for a 100 MW wind farm in North America with 600 turbines of 750 kW each. Field data from Germany (Faulstich *et al.*, 2008) indicate approximately six failures per year and restoration times ranging from 60 hours to a few weeks. As a result, O&M accounts

---

The journal model is *IIE Transactions*.

for 20-47.5% of the wholesale market price (Wiser and Bolinger, 2008). Considering that most turbines in the US were installed in the past 10 to 15 years, wind facilities are still operating in their relatively reliable period. In the next few years as turbine components near the end of their design life cycles, it is expected that failure rates of wind power facilities will soar exponentially, as will the costs of O&M.

I.2. Overview of wind farm operations and maintenance

Wind farm operators perform scheduled maintenances (SchMs) on a regular basis. However, since turbines are typically subjected to irregular loading (Leite *et al.*, 2006)), the deterioration progress of individual components often differs considerably. For this reason alone, SchMs may result in unnecessary visits, an inability to address unexpected failures in a timely fashion, etc.

To minimize O&M costs, wind farm operators have come to understand that condition-based maintenance (CBM) is essential to an effective maintenance program (Zhang *et al.*, 2009). For example, condition-based monitoring equipment (installing sensors inside turbines) provides diagnostic information about the health of components. Using such data helps wind farm operators to estimate the deterioration progress that may lead to major failure or consequential damage and establish appropriate maintenance in advance.

The CBM provides abundant information, but it cannot solve the uncertainty issue perfectly (Ding *et al.*, 2007). "Noise" can interfere with gathering measurement data, and a specific value of monitoring data can come from the different conditions of the target system. More importantly, fault diagnosis based on sensor measurements is nontrivial, because wind turbines operate under non-steady and irregular operating conditions. Given that it is difficult to determine the exact state of turbine

components forces operators to estimate the actual state in a probabilistic sense.

Several additional stochastic factors also need to be considered. One is the stochastic weather conditions that may constrain the feasibility of maintenance. Clearly, to maximize generation, wind facilities are built where the wind blows strongest. But climbing a turbine during wind speeds of more than 20 meters per second (m/s) is not allowed; when speeds are higher than 30 m/s, a site becomes inaccessible (McMillan and Ault, 2008). Moreover, some work takes days (and even weeks) to complete due to the physical difficulties of repairing or replacing components. The relatively long duration of a repair session increases the likelihood of disruption by adverse weather. A study using a Monte Carlo simulation (Rademakers *et al.*, 2003b) found that turbine availability was only 85-94% in a 100-unit wind farm situated about 35 kilometers off the Dutch coast. The relatively low availability is due to the farm's poor accessibility which is, on average, around 60%. Another study (Bussel, 1999) found the availability of a wind farm was 76%. Some repairs also require long lead times for assembling maintenance crews and obtaining parts. Pacot *et al.* (2003) points out that it may take several weeks for critical parts, such as a gearbox, to be delivered. Ultimately, these and similar factors affect the revenue loss incurred during downtimes. Lost productivity becomes more significant when turbine unavailability occurs in the high wind seasons (Walford, 2006).

I.3. Research objective and outline

Due to uncertainty and stochastic issues, a properly timed and well-planned preventive maintenance strategy should be of great interest to the wind industry. Hence, we propose a new integrated framework for wind farm O&M. The framework includes optimization models that provide the decision support capabilities enabling cost-

effective planning, easily implementable real-time control for turbine management, and a discrete event-based simulation model that characterizes the dynamic operations of wind power systems.

Fig. 1 depicts the integrated framework proposed. Within the framework, the optimization models produce dynamic O&M strategies based on mathematical analysis. The simulation platform characterizes the behavior of large-scale wind power systems with hundred-plus turbines per wind farm. It allows profound insights in developing optimization models for O&M because of its platform for testing different operational strategies.

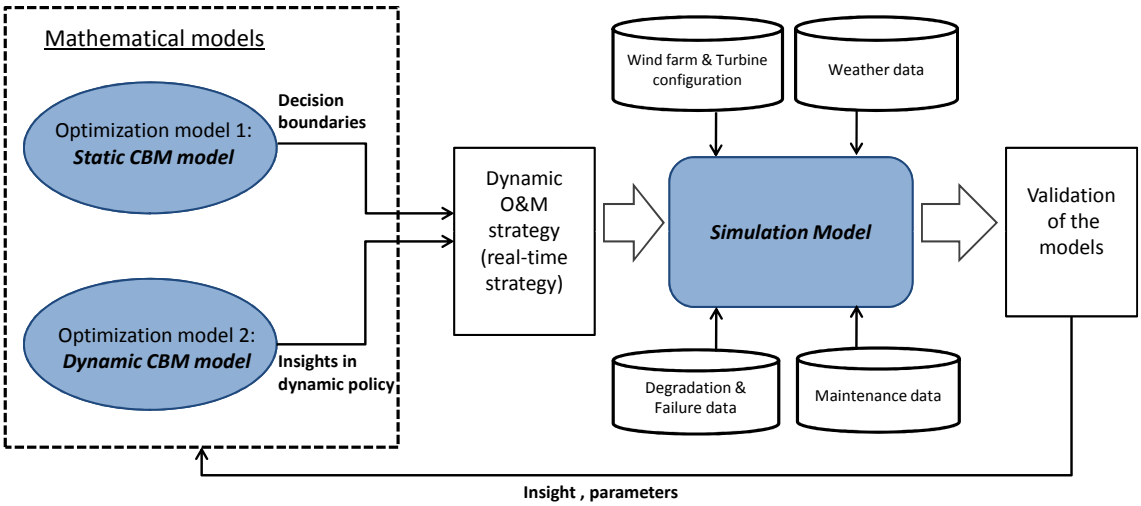


Fig. 1. Overall framework

I.3.1. Optimization models and solution tools for operation and maintenance

We develop two mathematical models to optimize maintenance activities. The models are based both on the internal condition of each turbine component and on

the external operating environments. The internal conditions include the degree of deterioration status (or health status) and the different failure modes associated with individual components. The external operating environment includes weather climate and required lead time to prepare repair resources. Although they may not be significantly related to the degradation or failure of a turbine component, they can impact the O&M costs and turbine availability (McMillan and Ault, 2008, Rademakers *et al.*, 2003a).

There are two types of measurements to estimate the internal condition of each turbine component: 1. inexpensive, but less reliable, remote sensing and diagnosis from general condition monitoring equipment, and 2. expensive, but more certain, on-site visit/observation (OB). Condition monitoring sensors can be run continuously but the information uncertainty must be handled with caution, and on-site observations must be integrated with planning other maintenance actions.

This dissertation addresses the problem of sensor information uncertainty by using a partially observed Markov decision process (POMDP), a sequential decision-making process used to control a stochastic system based on a system state (Lovejoy, 1987, Rosenfield, 1976). In a POMDP setting, the system condition cannot be observed directly, so that the condition is estimated in a probabilistic sense (Maillart, 2006, Maillart and Zheltova, 2007). Since sensors provide abundant yet uncertain data, a POMDP is aptly suited to optimize turbine maintenance activities.

Our optimization models aim at deciding the optimal decision strategy. The three types of actions are considered: preventive maintenance (PM); on-site visit and observation (OB); and when neither is needed, continue monitoring and take no action (NA). Regarding PM, we allow multiple repair levels that can bring an operating system to any state between the current state and an "as-good-as-new" state. We will examine the effects of each PM on costs, reliability and repair durations.

OB, as discussed earlier, differs from an automated, remote monitoring system. We define it as the infrequent, non-periodic on-site investigation that operators can take. OB is fulfilled by either dispatching a maintenance crew or, if technologically feasible, invoking more advanced smart sensors. Both options are generally costly, but presumably depict system conditions with a high confidence. We note that the co-existence of a cheap but unreliable remote monitoring and an accurate yet costly OB is unique to the wind industry.

With these different actions and the other several critical aspects, we propose two dynamic optimization models and their solution tools.

Model 1: This static, time-independent model with homogeneous parameters we term a *static CBM model*. The homogeneous parameters imply that the characteristics of weather conditions remain constant period by period. This relatively simple model allows us to characterize the solution structures and thus develop more efficient solution techniques. We analytically derive the optimal control limits for each action as a set of closed-form expressions. We provide the necessary and sufficient conditions under which preventive maintenance will be optimal and the sufficient conditions for other actions to be optimal.

The model can also incorporate several structural properties, such as the monotonicity of the optimal policy. We show that the structure of the optimal policy is similar to those studied in the previous POMDP literature, but our policy structure requires weaker assumptions. Optimality results for policy structures not previously proved in the literature are also presented. We examine the practical implications of these properties in wind turbine maintenance.

Model 2: This dynamic, time-dependent model with non-homogeneous parameters we term a *dynamic CBM model*. The time-varying parameters depend on prevailing weather conditions and exhibit considerable seasonal differences. Therefore,

the resulting strategy is adaptive to the operating environments.

In the dynamic CBM model we formulate the problem as a finite horizon POMDP model. The optimal policy is constructed from the evolution of the deterioration states of individual wind turbine components. We use a backward dynamic programming algorithm to solve the problem.

I.3.2. Discrete event simulation model

We use the discrete event modeling and simulation approach to build a generic simulation model for wind farm operations. Specifically, *discrete event system specification* (DEVS) formalism (Zeigler *et al.*, 2000) is used to derive the model that can be tailored to any real-world facility. DEVS is a formal modeling and simulation framework based on dynamical systems theory. We choose DEVS because it provides well-defined concepts for coupling components, hierarchical and modular model construction and an object-oriented substrate supporting repository reuse. Furthermore, its modular construction ability allows the modeler to design and construct each model independently for optimum efficiency. The models can interact with each other by adhering to well-defined protocols.

This effort involves the following subtasks: (a) building wind farm DEVS atomic models; (b) coupling the atomic models to create complex coupled models; (c) building the experimental frame (EF) to allow for a suite of simulation experiment choices; (d) computer implementation of the models; and (e) testing, verification and validation. The simulation platform enables profound framework and insights into the operation of large-scale wind power systems and provides a platform for testing and validating the different O&M strategies or policies. Pre-defined algorithms can be integrated into the simulation model to determine which O&M action to take for each of the possible states (or conditions) of the system. This approach not only

provides the clarity in wind farm operators' decision-making, but also computational efficiency in the simulation procedure.

We begin by dividing the simulation run of a suggested entire operating horizon into shorter time periods (for example, one week) in order to take the aforementioned internal/external stochastic factors into account. The weather model inside the simulation generates climate conditions such as wind speeds for each time (planning) period. We estimate the physical condition of turbine components based on recent sensor information. The data inputs give us a highly accurate schedule that maximizes the most feasible personnel/equipment allocations and maintenance operations for the planning period being considered. We can then shift the decision horizon to the next period to find the most feasible planning and scheduling, continuing the procedure until we reach the end of the decision horizon. Upon completing the simulation, the next step is to evaluate the implemented O&M strategy via several performance criteria. To illustrate the application of the simulation framework, we perform a case study based on field data from literature and industry. The results confirm our hypothesis that appreciable benefits can be expected when operators apply the CBM strategy.

Since the simulation model includes many critical aspects of wind farm operations, a broad array of potential applications in addition to the maintenance policy can be developed on this simulation platform, e.g., evaluating site viability (Wan *et al.*, 2003), grid connection, generation adequacy of wind power systems (Karki and Billinton, 2004), system reliability (Karki and Patel, 2009, Wen *et al.*, 2009), and so on.

I.4. Organization of this dissertation

The remainder of this dissertation is organized as follows. Chapter II surveys the various methods proposed in the literature for wind farm operations. We review several studies about the O&M aspects of turbines to attain general understanding. Then we review simulation studies for operations and their limitations. We also examine POMDP optimization models for general maintenance problems and explain the relationship between previous models and the models proposed in this study.

Chapter III discusses several modeling aspects relevant to wind farm O&M. We present the different choices of maintenance actions available to wind farm operators, the corresponding effects on system conditions, and the associated costs.

Chapter IV describes the static CBM model using a POMDP with static weather parameters. Several critical factors uniquely encountered in wind farm operations are incorporated in the model. We characterize the optimal policy after analyzing the structural properties of the presented model. We derive the closed expressions for each action to be optimal and give a computationally improved algorithm based on the developed decision rules.

Chapter V extends the static CBM model, incorporating more practical aspects of operations. We describe the dynamic CBM model with heterogeneous parameters by allowing dynamic weather conditions. A backward dynamic programming is devised to solve for the optimal policy numerically. We empirically demonstrate the performance of the model using a case with strong seasonality.

Chapter VI describes several critical components, such as wind turbine degradation, power generation, wind speed simulation, and maintenance, and how to couple such components to construct a generic simulation model. We conduct a case study by applying the model to a 100-unit facility with different O&M strategies, and compare

the performance of each strategy during the average life spans of the turbines.

Chapter VII introduces the integration framework to incorporate the optimization results in the simulation. We propose a real-time decision-making process based on the structural results garnered from the static CBM model and describe the preliminary results.

Chapter VIII summarizes our findings and offers suggestions for expanding our research to other classes of wind farm operations.

Appendix A gives detailed descriptions of the characteristics of the simulation model which may prove useful to those unfamiliar with DEVS formalism.

CHAPTER II

LITERATURE REVIEW

We first review several studies to understand the O&M aspects of wind turbine generators. Then, we review simulation studies for wind farm operations. We also examine several optimization models for general maintenance problems using a POMDP which incorporates the information from condition monitoring sensors.

II.1. Studies on operation and maintenance

II.1.1. Factors affecting wind farm operations

Several studies have examined critical factors which affect the O&M costs of wind generation. Pacot *et al.* (2003) discuss key performance indicators in wind farm management, and review the effects of several factors such as wind turbine age, turbine size, and location. Bussel (1999) presents an expert system to determine the availability of wind turbines and O&M costs; the goal is to find the most economical solution by striking a balance between front-loading costs invested for reliability enhancement and O&M costs.

Ribrant (2006) and Ribrant and Bertling (2007) review the different failure modes of turbine components and the corresponding consequences. For example, a failing gearbox can also lead to bearing failures, sealing problems, and oil system problems. According to Ribrant (2006), it can take several weeks to fix problems associated with bearing failures, partly because of the long lead time needed to have labor and heavy equipment in place, while oil system problems can usually be fixed within hours.

II.1.2. Condition-based maintenance techniques and benefits

Insightful review of the recent CBM for turbines is provided by Caselitz and Giebhardt (2005). Vibration analysis is the primary monitoring technique used for gearbox fault detection (Khan *et al.*, 2005). Other common monitoring systems include: measuring the temperature of bearings, lubrication oil particulate content analysis, and optical strain measurements (Ribrant, 2006).

A few studies attempt to quantify the benefits of CBM in the wind power industry. McMillan and Ault (2008) evaluate the cost-effectiveness of CBM via Monte Carlo simulations. They employ several probabilistic models to accommodate uncertainties which are incorporated in their Monte Carlo simulations to capture the complex processes. Through simulating various scenarios with different weather problems, down-time duration, and repair costs, they show that operators can gain economic benefits for onshore turbines by adopting specific CBM strategy. One would expect more appreciable benefits for offshore wind turbines since the repairs of those turbines are more costly and taking maintenance actions faces more constraints. Similarly, Nilsson and Bertling (2007) present an asset life-cycle cost analysis by breaking all maintenance costs into several cost components. They analyze the benefits of CBM with a case study of two wind farms in Sweden and the UK.

II.1.3. Mathematical models for optimal strategies

There are two categories of mathematical models. The first uses statistical methods to identify the optimal repair time based on failure statistics. Using statistical approaches, Andrawus *et al.* (2007) employ the Weibull distribution to model the failure pattern of each component before deciding on the optimal replacement cycle for each component. According to their case studies of 600 kW horizontal axis

turbines, a gearbox should be replaced every six years and a generator every three years to minimize total maintenance costs. Similarly, Hall and Strutt (2007) develop probabilistic failure models for assessing component reliability. Both studies consider the average aging process of the components, but do not capture the degradation behavior of each individual component.

The second category uses Markov models to analyze the aging behavior of wind components. Notably, Markov models are in wide use because of their flexibility and popularity in many industrial applications (Billinton and Li, 2004, Hoskins *et al.*, 1999, Jirutitijaroen and Singh, 2004, Qian *et al.*, 2007, Welte, 2009, Yang *et al.*, 2008). To date, however, Markov models have rarely been applied to the wind industry. Sayas and Allan (1996) evaluate the generation availability of wind farms using a Markov model, in which wind turbine condition is categorized by two simple states: up or down. A simple failure model is considered, i.e. the time to failure of each generator including both turbines and conventional generators is assumed to be exponentially distributed, and the mean time to failure (MTTF) of each generator is estimated from historical data. Based on historical data about wind turbine failures, the transition probabilities between the two states are also obtained.

McMillan and Ault (2008) further categorize the states of a wind turbine condition in more detail by considering the individual state of critical components like gearboxes and generators. Their Markow model incorporates an intermediate state to represent component degradation behavior for a gearbox. Condition monitoring equipment to evaluate the system state is employed, but it is assumed that the condition monitoring equipment exactly reveals the degradation status of each turbine component.

II.1.4. Limitations of existing models

Most existing wind farm O&M models neglect the critical factors addressed in Chapter I. For example, most mathematical models described in Section II.1.3 only consider the degradation condition of turbine components, and omit other exogenous factors such as weather constraints and lead time to prepare resources upon a failure. Moreover, the decisions for repairing wind turbines are based on average aging behavior, but the real-time sensory information which reflects the actual condition of each turbine component is not integrated. Only McMillan and Ault (2008) consider the sensory information in their aging model, but they do not take into account the uncertainty associated with the sensor information.

Finally, the existing studies only consider preventive repairs as a decision alternative. As discussed in Section I.3.1, on-site investigation (OB) to examine the exact condition of wind turbine components is costly, but provides the most accurate information about component condition. Therefore, OB must be planned carefully when planning other maintenance actions.

II.2. Simulation model for wind farm operations

In this section, we first survey the simulation studies directly related to O&M. Then, we review studies which are not specific to wind farm O&M, but from which we hope to gain insights about reliability and the cost issues of wind energy within the entire electric power system. These studies provide the broad applicability of our simulation model to other classes of problems associated with wind power operations.

II.2.1. Simulation models related to wind farm operation and maintenance

Rademakers *et al.* (2003b) describe a Monte Carlo simulation model for operations and maintenance of offshore wind farms, developed by Delft University of Technology (TU-Delft). They illustrate the features and benefits of the model using a case study of a 100 MW wind farm. The model simulates the operation aspects over a period of time by considering several critical factors for performing repair actions, such as turbine failures and weather. The failures of turbine components are generated stochastically, based on the relevant statistics such as MTTF and reliability distributions. Weather conditions are realized with the given summer and winter storm percentages at the specific site. The model further categorizes different failure modes and the corresponding repair actions. For example, the first category of the failure mode requires the replacement of rotor and nacelle with external crane; the second failure mode requires replacement of large components with internal crane, and so on. The failure rates of the individual components are distributed over four maintenance categories. The model only considers corrective maintenance, and the simulation results indicate that the revenue losses account for 55% of the total maintenance costs, mainly due to the long lead times to prepare parts and the waiting time until favorable weather conditions are met. Similar studies appear in Bussel (1999), Rademakers *et al.* (2003a) and Hendriks *et al.* (2000).

McMillan and Ault (2008) quantify the cost-effectiveness of CBM by comparing the performance of different maintenance polices. Several probabilistic models are employed to introduce uncertainties. They use an autoregressive time series analysis to generate wind speeds and consider weather constraints when performing repair actions.

Simulations are also used for the validation purpose of various O&M approaches. In Andrawus *et al.* (2007), the suggested strategy resulting from their statistical model is evaluated by using Monte Carlo simulations. They assess the reliability, availability, and maintenance costs by simulating a wind farm with turbines over a period of four years using a commercial software called ReliaSoft BlockSim-7 (ReliaSoft BlocSim-7 software, 2007). Similarly, Hall and Strutt (2007) develop probabilistic failure models for component reliability using Monte Carlo simulation combined with statistical analysis.

II.2.2. Other simulation models

Karki and Billinton (2004) use a Monte Carlo simulation to help determine appropriate wind power penetration in an existing power system from both reliability and economic aspects. The generating system is divided into subsystems of wind turbine generators and conventional generators. The power output generated from the wind system is combined with the capacity of the conventional system to create the generation model for the entire power system. In simulating wind speeds to determine the generated power from the wind turbines, the authors use an autoregressive moving average (ARMA) time series model. A simple failure model is considered in this study; i.e. the time to failure of each generator including wind turbines and conventional generators is assumed to be exponentially distributed, and the MTTF of each generator is estimated from historical data. Based on a case study of a typical small power generating system, the authors present the procedure to help determine an appropriate wind penetration level in a power system with both reliability and cost criteria.

Karki and Patel (2009) extend the above study to determine appropriate transmission line size and evaluate the reliability of the combined wind generation and

transmission systems. Several other studies also use Monte Carlo simulation to evaluate the reliability and/or availability of hybrid power systems including wind power (Di Fazio and Russo, 2008, Leite *et al.*, 2006, Ravindra and Prakash, 2008). For a detailed review of recent reliability assessment studies on wind power, also see Wen *et al.* (2009).

II.2.3. Limitations of existing simulation models

Discrete event simulation and modeling (Law and Kelton, 1997, Zeigler et al., 2000) is an operations research technique that has been used in the past to look at the characteristics of many applications, yet studies applying it to wind energy are scarce. To the best of our knowledge, this dissertation presents the first discrete-event simulation model specific to wind farm operations.

In discrete event simulation, the operation of each model is represented as a chronological sequence of events which occur at discrete time instances and can alter the system state (Miller *et al.*, 2009). Our simulation model also belongs to the category of a Monte Carlo simulation, in the sense that it uses random number generators to characterize the stochastic aspects of wind farm operations. A major difference between our proposed model and existing models is that in the latter time evolution is unimportant, and the focus is to obtain lump sum estimates for performance measures. On the contrary, our model enables operators to gain a detailed knowledge of the lifetime evolution of wind power systems in addition to gathering performance measures.

Moreover, the current simulation models are generally oversimplified without sufficient granularity representing wind farm operations. Most of them assume independence, yet in reality there is a high dependency of wind turbines in power generation. There is a spatial correlation of wind speeds at wind turbine sites, and as

a result, generation is also correlated among the turbines. In addition, a well-designed model must consider the elevation information and the morphology of the terrain at a turbine site, because these factors affect power production.

Perhaps most critical is the lack of decision-making ability inside existing simulation models, Simply put, there is no integrated framework for wind farm operations in which the simulations can interact with decision-making modules during simulation runs.

II.3. Optimization models using partially observed Markov decision processes (POMDPs)

Recently several mathematical models have been introduced which incorporate information from CBM. Although they are not specific to turbine maintenance, we can gain insights about the utilization of CBM data.

II.3.1. Static operating environment

Maillart (2006) uses POMDPs to adaptively schedule the observation and to decide the appropriate maintenance actions based on the state information from CBM. In her study, the system is assumed to undergo a multi-state Markovian deterioration process with a known and fixed transition probability matrix. Gebraeel (2006) integrates the real-time sensory signals from CBM with a population-specific aging process to capture the degradation behavior of individual components. The author updates the remaining life distributions of individual components in a Bayesian manner. Similarly, Ghasemi *et al.* (2007) represent a system's deterioration process using the average aging behavior provided by the manufacturer (or from survival data) and the system utilization that can be diagnosed by CBM data. They formulate the problem via a

POMDP and derive optimal policies using dynamic programming.

Several studies examine the structural properties of POMDP maintenance models (Lovejoy, 1987, Maillart, 2006, Maillart and Zheltova, 2007, Ohnishi *et al.*, 1986, Rosenfield, 1976, Ross, 1971). Although they choose different state definitions and cost structures, they establish a monotonic "At-Most-Four-Region" (AM4R) structure. AM4R implies that along ordered subsets of deterioration state spaces, the optimal policy regions are divided into four regions at most in the following order: taking no action → taking observation to perfectly identify the physical condition of a system → taking no action → preventive maintenance. For example, Ohnishi *et al.* (1986) prove similar results for the problem where a system is monitored incompletely in discrete decision epochs, but taking observation perfectly reveals the condition of a system at some cost (for detailed reviews of these AM4R studies, see Maillart (2006)).

II.3.2. Stochastic operating environment

Most maintenance studies in the literature to date only consider static environmental conditions. Very few quantitative studies exist for systems operating under stochastic environments. Thomas *et al.* (1991) investigate the repair strategies to maximize the expected survival time until a catastrophic event occurs in an uncertain environment. They consider the situation where a system should be stopped during inspection or maintenance action. If specific events, termed "*initiating events*", take place when a system is down or being replaced, they are noted as catastrophic events; examples given are military equipment or hospital systems. Thomas *et al.* (1991) also show that similar AM4R structural results hold for a simple system in which the system state takes only the binary values, operating or failed. Kim and Thomas (2006) extend the problem where the multiple environmental situations are assumed to follow Markovian process. However, the criteria in both studies are designed to maximize the expected

time until a catastrophe occurs. In sum, they focus on short-term availability, whereas we are interested in minimizing long-term total costs.

II.3.3. Two proposed optimization models

We devise two multistate POMDP models to capture the degradation process of wind turbines and to decide the optimal maintenance strategies. The presented models extend the model introduced in Maillart (2006) by incorporating the unique characteristics of turbine operations. For example, to represent stochastic weather conditions, we apply the initiating events idea described above, since harsh weather conditions delay repair processes and cause non-negligible revenue losses. Other characteristics included are long lead times after unplanned failures and the resulting production losses.

For the static CBM model with homogeneous parameters, we show that the optimal decision rules are composed of control limit polices. Previous POMDP studies show the existence of an optimal control limit for preventive maintenance action (Maillart and Zheltova, 2007, Ohnishi *et al.*, 1986), i.e. if preventive maintenance is an optimal action for a system it is also an optimal action for a more deteriorated system. These studies do not provide the exact value of the control limit. In contrast, this dissertation derives the closed-form conditions for each action to be optimal. Establishing closed-form expressions helps us to efficiently find optimal strategies that produce significant speedups in high dimensional problems.

We also show that the static CBM model still holds the well-known monotonic AM4R policy structure under weaker assumptions than Maillart (2006). Since the model can be generalized by varying the parameter values, the AM4R policy structure we present can be applied not only to wind turbines, but also to other general aging systems. Additionally, we demonstrate the conditions under which the optimal policy

structure becomes a more appealing monotonic "At-Most-Three-Region" (AM3R) structure. AM3R implies that there are at most three optimal policy regions along ordered subsets of the deterioration state spaces in the following order: taking no action →taking observation → preventive maintenance. There is no second "taking no action" region in a AM3R structure, which there is one in the AM4R structure. This simpler structure is more intuitive and easier to implement.

In the dynamic CBM model, we gain insights about adapting the repair strategy based on changing operating environments. We solve the problem by a backward dynamic programming. Due to the heterogeneity of weather parameters, we cannot derive the structural properties as we established for the static CBM model. However, as discussed in Chapter VII, we suggest a real-time algorithm to find the approximate decision rules for the optimal policy based on the results of the static CBM model. In the real-time algorithm, the most updated weather information is applied to decide the proper maintenance policy. We provide the preliminary results to empirically validate the real-time algorithm using a case study.

CHAPTER III

MODELING THE OPERATION AND MAINTENANCE IN WIND FARMS

In this chapter, we examine several modeling aspects relevant to the wind farm operations. We also consider the different choices of maintenance action that the wind farm operators can take and the corresponding effects on the system condition and the associated costs. It is assumed that the wind farm operators make maintenance decisions in discrete time.

III.1. Modeling aspects in wind farm operations

III.1.1. Different failure modes

Wind turbine components experience different failure modes, leading to the different failure consequences. Each failure mode determines what type of parts/crew is required, which in turn determines the costs, lead time and repair time. Accordingly, the costs of corrective maintenance (CM) and the downtime due to the occurrence of a turbine failure could vary for different failure modes. We assume that a system can experience L types of failures.

III.1.2. Partial information about a system

Suppose that the deterioration levels of an operating system are classified into a finite number of conditions $1, \cdots, M$ and that there are L different types of failures. Then, the system condition can be categorized into a series of states, $1, \cdots, M + L$. State 1 denotes the best condition like "new", and state M denotes the most deteriorated operating condition before a system fails. State $M + l$ reflects the l^{th} failed mode,

$l = 1, \cdots, L$. Let us call $S_0 = \{1, \cdots, M + L\}$ the *original state space*.

In reality, the physical condition of a turbine component is not known exactly, but may be estimated from the condition monitoring sensor signals. Estimations rarely reveal perfectly the system conditions and health status due to a wide variety of reasons, such as imperfect models linking measurements to specific faults, as well as noises and contaminations in sensor signals (Ding *et al.*, 2007). One way to characterize the information from the sensor signals is to specify a probability vector about the actual underlying condition. A common treatment of the information uncertainty under the POMDP setting is to define a state as a probability distribution, representing one's belief over the corresponding true state. As such, we define the state of the system as the following probability distribution

$$\pi = [\pi_1, \pi_2, \cdots, \pi_{M+L}], \tag{3.1}$$

where π_i, $i = 1, \cdots, M + L$ is the probability that the system is in deterioration level i. π is commonly known as an information state in the literature (Maillart and Zheltova, 2007). Then, the state space under the POMDP setting becomes

$$S_1 = \{[\pi_1, \pi_2, \cdots, \pi_{M+L}]; \sum_{i=1}^{M+L} \pi_i = 1, 0 \leq \pi_i \leq 1, i = 1, \cdots, M + L\} \tag{3.2}$$

Let us call S_1 the *partially observed state space*.

When one of the elements in the information state is *one* and other elements are *zero*, the state is called the extreme state, denoted by e_i, $i = 1, \cdots, M + L$, where $e_i = [0, \cdots, 1, \cdots, 0]$ is $(M+L) \times 1$ dimensional row vector with a 1 in the i^{th} position and 0 elsewhere. In other words, e_1 denotes the best condition like an "as-good-as new" condition, e_M is the most deteriorated condition, and e_{M+l}, $l = 1, \cdots, L$ denotes the l^{th} failure mode. These extreme states reveal the system's condition perfectly.

Note that $\sum_{i=1}^{M} \pi_i = 1$ for an *operating* system since wind turbines no longer

operate upon failures. When a system fails with the l^{th} failure mode, the state becomes e_{M+l}.

III.1.3. Markovian deterioration

In this study, we choose a Markov model to represent the aging behavior of a system. When a system undergoes Markovian deterioration, the current state is transited to another state according to a transition probability matrix, $P = [p_{ij}]_{(M+L)\times(M+L)}$. P consists of the four submatrices as follows:

$$P = \begin{bmatrix} P_A & P_B \\ 0_{L\times M} & I_{L\times L} \end{bmatrix}, \tag{3.3}$$

where P_A denotes an $M \times M$ transition matrix from an operating state to another operating state, and P_B is an $M \times L$ transition matrix from an operating state to one of the failure states. $0_{L\times M}$ is an $L \times M$ zero matrix, whereas $I_{L\times L}$ is an identity matrix. Together, $0_{L\times M}$ and $I_{L\times L}$ matrices reflect the fact that once the system fails, it cannot return to any operating state on its own but remains at the same failure state unless a maintenance action is taken. In many practical applications, P is an upper-triangular matrix where the lower off-diagonal elements are zero because a system cannot improve on its own. Fig. 2 illustrates the state transitions with an upper-triangular matrix P in the original state space S_0 .

Suppose that the current information state of an operating system is π and no action (NA) is taken. The probability that the system will still operate until the next decision point is $R(\pi) = \sum_{i=1}^{M}\sum_{j=1}^{M}\pi_i p_{ij}$. People call this probability as the *reliability* of the system (Maillart, 2006). Based on the law of conditional probability (Maillart, 2006), the information state after the next transition, given that the system

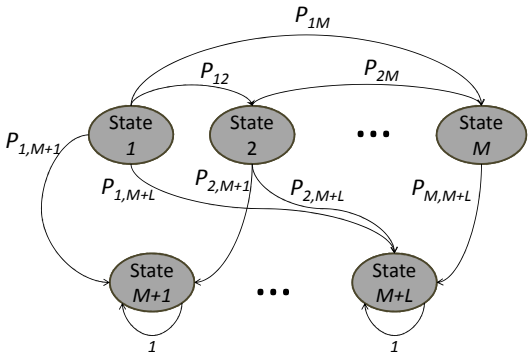


Fig. 2. State transition diagram in the original state space

is not yet failed, is

$$\pi'_j(\pi) = \begin{cases} \frac{\sum_{i=1}^{M} \pi_i p_{ij}}{R(\pi)}, & j = 1, 2, \cdots, M \\ 0, & j = M+1, \cdots, M+L. \end{cases} \tag{3.4}$$

As such, the system is transited to the next state $\pi'(\pi) = [\pi'_1(\pi), \cdots, \pi'_M(\pi), 0, \cdots, 0]$ with probability $R(\pi)$.

As an additional note, sensor information is not reflected in the state transition in (3.4) because the uncertainty and bias from sensor outputs may contaminate the information state further. How the sensor information should be used in updating the information state has been addressed in several studies (Porta *et al.*, 2006, Qian *et al.*, 2007, Spaan and Vlassis, 2005). But the method and analysis is not straightforward and remains as an open question yet. Since this issue is out of scope of this dissertation, we simply apply the equation (3.4) for modeling the state transition.

If the system fails and results in the l^{th} failure mode with probability $H_l(\pi) = \sum_{i=1}^{M} \pi_i p_{i,M+l}$, the state becomes e_{M+l} in the next period. And, the total probability that the system fails until the next period is $H(\pi) = 1 - R(\pi) = \sum_{j=M+1}^{M+L} H_l(\pi)$, which is called the *hazard rate* of the system. Fig. 3 illustrates the state transition diagram in the partially observed state space S_1 without any maintenance interruption.

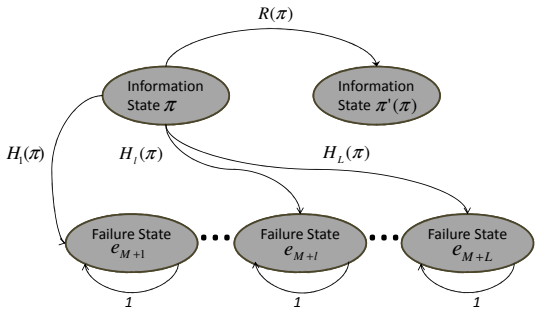


Fig. 3. State transition diagram in the partially observed state space

III.2. Maintenance actions

III.2.1. Corrective maintenance

According to Walford (2006), the portion of the corrective maintenance costs is between 30% and 60% of the total O&M costs. Not only do the direct costs (to fix the failed components), but the indirect costs such as revenue losses also contribute considerably to the corrective maintenance costs. This is mainly the result of a typically long downtime, due to usually restricted accessibility to a wind farm and limited availability of parts and crew (Rademakers *et al.*, 2003a).

Upon a failure with the l^{th} failure mode, parts are ordered and crews are arranged, which supposedly takes $\lambda(l)$ lead time. When all of the parts and crew are available, and if the weather conditions are good enough to allow the repair work to go ahead, the crew carry out a CM for the l^{th} failure mode (namely, $CM(l)$) for $\mu(l)$ repair periods at cost $C_{CM(l)}$ (note: $\lambda(l)$ and $\mu(l)$ take non-negative integer values, meaning 0 period, 1 period, 2 periods and so on). If the prevailing weather conditions are not good enough, however, the crew must wait until the weather conditions permit a repair. Let $W_{CM(l),n}$ represent the probability that the prevailing weather conditions during the n^{th} period are harsh, and CM for the l^{th} failure mode is thus prohibited.

Without loss of generality, we order the failure states such that a higher index implies a more serious failure mode. We assume that major repairs that fix serious problems take one full period (i.e. $\mu(l) = 1$), whereas the repair time for minor problems is negligible compared to the duration of a period (i.e. $\mu(l) = 0$). Understandably, major repairs require that the weather conditions stay permitting for the whole repair period, and also require costlier resources and longer lead time than minor repairs. Therefore, we have $W_{CM(l),n} \leq W_{CM(l'),n}$, $C_{CM(l)} \leq C_{CM(l')}$, and $\lambda(l) \leq \lambda(l')$ for $l \leq l'$.

Unless the repair is completed, wind turbines can no longer be operated after a failure, causing τ_n revenue losses at period n. Note that τ_n could be a time-dependent parameter, varying season by season. After the repair, the system is renewed to an as-good-as-new state. Fig. 4 illustrates the repair process after a failure occurs. The figure illustrates the repairing process after a major failure where $\mu(l) = 1$. For a minor failure, the process would be similar except $\mu(l) = 0$

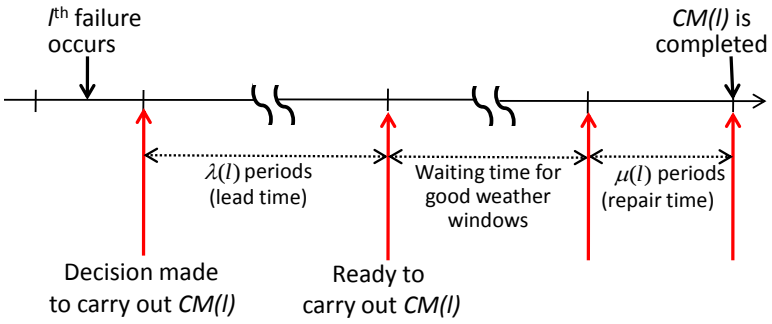


Fig. 4. Corrective maintenance after a failure with the l^{th} failure mode

III.2.2. Preventive maintenance

PM is the action to repair the system that has deteriorated but not yet failed (Chattopadhyay, 2004). The PM's are divided based on how system condition can be

improved with maintenance efforts. Recall that the condition of an *operating* system in this study is modeled by M discrete levels, which suggests that there can be at most $M-1$ choices for the PM actions, namely, $PM(1), \cdots, PM(M-1)$, where $PM(m)$ denotes the PM action which repairs the system to the state e_m at cost $C_{PM(m)}$. For example, choosing $PM(1)$ corresponds to performing a major repair such as overhaul, which returns the system to an as-good-as-new state, e_1. On the other hand, $PM(M-1)$ spends the least efforts to bring the system state to e_{M-1}. Accordingly, $C_{PM(m)} \geq C_{PM(m')}$, $\forall m \leq m'$.

Depending on which PM level is chosen, the repair time and the requirements for weather conditions may differ, and consequently, the production loss during a repair can be different. If the weather becomes harsh during a repair, the crew have to hold the repair work until the weather returns to good conditions. Let $W_{PM(m),n}$ represent the probability that the weather conditions at period n do not allow $PM(m)$ to be performed, $m = 1, \cdots, M-1$. Then, we have $W_{PM(m),n} \geq W_{PM(m'),n}$ for $m \leq m'$.

III.2.3. Observation

Through the remote monitoring system, the wind farm operators can attain the partial (and imperfect) information about the system condition, while OB is the action to evaluate the system's exact deterioration level at cost C_{OB}. The information state after an OB reverts to one of the extreme states e_i, $i = 1, \cdots, M$, where e_i is defined earlier in Section III.1.2. After an OB, the decision maker will choose an adequate maintenance action in that same decision period, based on the updated information state.

III.2.4. No action

NA is the action to continue the operation without any intervention. With this action, the system undergoes deterioration according to a known transition probability matrix, $P = [p_{ij}]_{(M+L)\times(M+L)}$ in (3.3). Given the current information state π, under NA, the system will transit to the next state $\pi'(\pi)$ in (3.4) in the next decision point with probability $R(\pi)$ or fail with the l^{th} failure mode with probability $H_l(\pi)$.

III.3. Illustrative example

Suppose that the operating system condition can be categorized into three different aging levels, namely, *normal*, *alert* and *alarm* condition, respectively. π_1, π_2 and π_3 denote the probability that the system condition is *normal*, *alert* and *alarm*, respectively. Fig. 5 illustrates the partially observed state space for an operating system. In the figure, the X-axis denotes the *alert* probability π_2, whereas the Y-axis is the *alarm* probability π_3. Since $\pi_1 + \pi_2 + \pi_3 = 1$, the origin $(0, 0)$ implies the best condition e_1. The state space is defined as the triangle surrounded by the X-axis, Y-axis and $\pi_2 + \pi_3 = 1$. Note that $\pi_2 \geq 0$, $\pi_3 \geq 0$ and $\pi_2 + \pi_3 \leq 1$. Therefore, all states can only fall inside the triangular area, as shown in Fig. 5.

The *area A* in the upper-left corner of the triangle depicts the states corresponding to seriously deteriorating conditions. The states belonging to this area might need remedies such as preventive repairs to avoid a catastrophic failure in the near future. On the other hand, the *area B* in the lower-left corner implies the healthy conditions, which might not need any repair action. The states outside these two areas are those whose aging conditions are in-between and the information about the health status of the system is obscure. Therefore, OB might be necessary when the system state belongs to this in-between area.

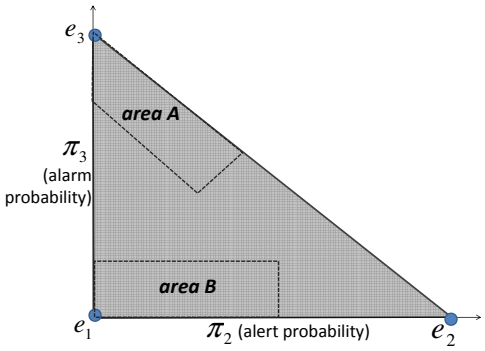


Fig. 5. Partially observed state space for an operating system

CHAPTER IV

STATIC CBM MODEL: A POMDP MODEL WITH HOMOGENEOUS PARAMETERS

In this chapter we formulate the wind turbine maintenance problem using a POMDP with homogeneous parameters. We introduce the existing algorithm to numerically solve the problem and in later sections, we will present a computationally improved algorithm after analyzing the structural properties of the presented model.

IV.1. Model formulation

Let $V_n(\pi)$ denote the minimum expected total cost when n periods are left to the terminal period (or the total cost-to-go) and the current state is π. At each decision epoch, there are $M+1$ possible action alternatives: NA, $PM(1), \cdots, PM(M-1)$, and OB. In this model, we assume the weather characteristics remain constant across the decision horizon. That is, all weather related parameters are constant, and we set $W_{CM(l)} = W_{CM(l),n}$, $W_{PM(m)} = W_{PM(m),n}$ and $\tau = \tau_n$ for $\forall l, m, n$.

When NA is selected at the current state π, the total cost-to-go can be formulated as follows:

$$NA_n(\pi) = \sum_{l=1}^{L} \left(\lambda(l) \cdot \tau + CM_{n-\lambda(l)-1}(e_{M+l})\right) H_l(\pi) + V_{n-1}(\pi'(\pi))R(\pi) \tag{4.1}$$

where

$$\begin{aligned} CM_n(e_{M+l}) =& W_{CM(l)} \left(\tau + CM_{n-1}(e_{M+l})\right) + \\ & (1 - W_{CM(l)})(\tau \cdot 1(\mu(l) = 1) + C_{CM(l)} + V_{n-\mu(l)}(e_1)) \end{aligned} \tag{4.2}$$

Here, $1(\cdot)$ is the indicator function. Under NA, the system could either end up with

the l^{th} failure mode with probability $H_l(\pi)$, $l = 1, \cdots, L$, or, transit to the next state $\pi'(\pi)$ with probability $R(\pi)$. In (4.1), the first term $\lambda(l) \cdot \tau$ is the total revenue losses during the lead time upon a system failure with l^{th} mode. $CM_n(e_{m+l})$ in (4.2) reflects the CM costs for the l^{th} failure mode. The first component in (4.2) is the expected costs caused by delays due to harsh weather conditions, which would occur with probability $W_{CM(l)}$. The second component indicates the repair costs under good weather conditions. Note that $\tau \cdot 1(\mu(l) = 1)$ in (4.2) specifies the revenue losses during a major repair that takes one full period. After the repair, the system condition is restored to the best condition e_1.

Next, let us consider the actions of PM. $PM(m)$ action, $m = 1, \cdots, M-1$ improves the system condition to the state e_m. In this static CBM model, we assume that all of the preventive repairs take one full period and if the weather conditions become harsh during the repair, the job has to be halted and will be resumed in the next period. The following formulation in (4.3) is the total cost-to-go for $PM(m)$ for $m = 1, \cdots, M-1$:

$$PM_n(m) = W_{PM(m)}(\tau + PM_{n-1}(m)) + (1 - W_{PM(m)})(\tau + C_{PM(m)} + V_{n-1}(e_m)) \tag{4.3}$$

Finally, we model the action of OB. The observation costs are divided into the direct costs to inspect the system and the post maintenance costs after the system condition is evaluated precisely. The following $OB_n(\pi)$ and $Post_n(\pi)$ together represent that after each observation at cost C_{OB}, the state is updated to e_i with probability π_i and then we choose the least costly action in the same decision period, among NA or $PM(m)$, $m = 1, \cdots, M-1$.

$$OB_n(\pi) = C_{OB} + \sum_{i=1}^{M} Post_n(e_i)\pi_i \tag{4.4}$$

where

$$Post_n(e_i) = min\{NA_n(e_i), PM_n(1), \cdots, PM_n(M-1)\} \tag{4.5}$$

Note that OB cannot be optimal at the extreme points e_i, $i = 1, \cdots, M$ because $OB_n(e_i)$ is always greater than $min\{NA_n(e_i), PM_n(1), \cdots, PM_n(M-1)\}$ when $C_{OB} > 0$.

Now, the optimal value function can be written as follows:

$$V_n(\pi) = min\{NA_n(\pi), PM_n(1), \cdots, PM_n(M-1), OB_n(\pi)\} \tag{4.6}$$

Solving the optimization in (4.6) gives the the optimal decision rule $\delta_n^S(\pi)$ at the current state π where the superscript "S" implies a *static* policy. Here, $\delta_n^S(\pi)$ will take one of the possible actions, NA, $PM(1), \cdots, PM(m-1)$, OB, specifying the best action selection when the system occupies the state π at a specified decision epoch n.

IV.2. Limiting behavior

Since the system is renewed after CM or the system condition is improved after $PM(m)$ as long as weather conditions are good, the model is unichain for $0 \leq W_{CM(l)} < 1$ and $0 \leq W_{PM(m)} < 1$, $\forall l, m$ (Maillart, 2006). That is, the transition matrix corresponding to each action consists of a single recurrent class. For these kinds of problems, Puterman (1994) shows that $V_n(\pi)$ approaches a line with slope g and intercept $b(\pi)$ as n becomes large, which is,

$$\lim_{n\to\infty} V_n(\pi) = n \cdot g + b(\pi) \tag{4.7}$$

Here, g denotes the average cost per unit time under the optimal policy, $b(\pi)$ is the bias, or the relative cost when the information state starts from π.

To obtain g and $b(\pi)$, we should take limit for the cost-to-go associated with each action. First, taking limit of $NA_n(\pi)$ and then applying (4.7) yields

$$\lim_{n\to\infty} NA_n(\pi) = \lim_{n\to\infty} \sum_{l=1}^{L} \left(\lambda(l)\cdot\tau + CM_{n-\lambda(l)-1}(e_{M+l})\right) H_l(\pi) + V_{n-1}(\pi'(\pi))R(\pi) \tag{4.8}$$

$$= \sum_{l=1}^{L} \left(\lambda(l)\cdot\tau + (n-\lambda(l)-1)g + b(e_{M+l})\right) H_l(\pi) + \left((n-1)g + b(\pi'(\pi))\right) R(\pi) \tag{4.9}$$

$$= ng + \sum_{l=1}^{L} \left(\lambda(l)(\tau-g) + b(e_{M+l})\right) H_l(\pi) + b(\pi'(\pi))R(\pi) - g \tag{4.10}$$

Let us denote the bias associated with action NA by $b_{NA}(\pi)$. Then, we can define $b_{NA}(\pi)$ as follows:

$$b_{NA}(\pi) = \sum_{l=1}^{L} \left(\lambda(l)(\tau-g) + b(e_{M+l})\right) H_l(\pi) + b(\pi'(\pi))R(\pi) - g \tag{4.11}$$

Here, $b(e_{M+l})$ can be obtained by applying the same technique to $CM_n(e_{M+l})$ in (4.2) as follows:

$$b(e_{M+l}) = C_{CM(l)} + b(e_1) + \begin{cases} \frac{W_{CM(l)}(\tau-g)}{1-W_{CM(l)}} & \text{if } \mu(l)=0 \\ \frac{(\tau-g)}{1-W_{CM(l)}} & \text{if } \mu(l)=1 \end{cases} \tag{4.12}$$

Since $b(\pi)$ is the relative difference in total cost that results from starting the process in state π instead of any other state, Puterman (1994) suggests to set $b(\pi^0)=0$ for an arbitrary π^0. Intuitively, we set $b(e_1)=0$ in (4.12).

Similarly, let us denote the bias associated with $PM(m)$ action by $b_{PM(m)}(\pi)$.

That is,

$$\lim_{n\to\infty} PM_n(m) = n \cdot g + b_{PM(m)}(\pi) \tag{4.13}$$

Applying (4.13) to (4.3) in both sides gives

$$\begin{aligned} n \cdot g + b_{PM(m)}(\pi) =& W_{PM(m)}(\tau + (n-1)g + b_{PM(m)}(\pi)) + \\ & (1 - W_{PM(m)})(\tau + C_{PM} + (n-1)g + b(e_m)) \end{aligned} \tag{4.14}$$

Rearranging the above equation gives, for $m = 1, \cdots, M-1$,

$$b_{PM(m)}(\pi) = C_{PM(m)} + b(e_m) + \frac{(\tau - g)}{1 - W_{PM(m)}} \tag{4.15}$$

Let us now define the new maintenance costs which compound weather effects, lead time and production losses by C'_{CM} and C'_{PM}, respectively, as follows:

$$C'_{CM(l)} = C_{CM(l)} + \lambda(l) \cdot (\tau - g) + \begin{cases} \frac{W_{CM(l)}(\tau - g)}{1 - W_{CM(l)}} & \text{if } \mu(l) = 0 \\ \frac{\tau - g}{1 - W_{CM(l)}} & \text{if } \mu(l) = 1 \end{cases} \tag{4.16}$$

$$C'_{PM(m)} = C_{PM(m)} + b(e_m) + \frac{\tau - g}{1 - W_{PM(m)}} \tag{4.17}$$

Here, the new corrective cost, $C'_{CM(l)}$ consists of three terms, each of which has physical implications. The first term, $C_{CM(l)}$ is a direct repair cost. The second term is the revenue losses *minus* average maintenance costs during the lead time (Note that during a waiting time, no maintenance cost is incurred). The last part is the expected revenue losses during the repair delay due to weather constraints. For example, when $\mu(l) = 0$, the expected repair time is $\frac{W_{CM(l)}}{1 - W_{CM(l)}}$. Similarly, when the repair takes one period (i.e. $\mu(l) = 1$), the expected number of periods until the repair is finished is $\frac{1}{1 - W_{CM(l)}}$. During those periods, revenue losses incur. The new preventive maintenance cost, C'_{PM} can be interpreted likewise.

Also note that both $C'_{CM(l)}$ and $C'_{PM(m)}$ are increasing in $W_{CM(l)}$ and $W_{PM(m)}$, respectively. This implies that higher frequency of harsh weather conditions incurs higher repair costs. Here, we assume that $\tau \geq g$, i.e. the revenue per period is greater than, or equal to, the average cost. Therefore, the added costs due to an unplanned failure (i.e. $C'_{CM} - C'_{PM}$) arise from the following three factors: (1) increased repair costs (for doing CM), i.e. $C_{CM(l)} - (C_{PM} + b(e_m))$; (2) production losses caused by the waiting time to prepare resources after a failure, i.e. $\lambda(l) \cdot (\tau - g)$; and (3) increased possibility of repair delays due to more restricted weather requirements to carry out CM.

Substituting C'_{CM}, C'_{PM} into (4.11) and (4.15) simplifies the equations to

$$b(\pi) = min \begin{cases} b_{NA}(\pi) = \sum_{l=1}^{L} C'_{CM(l)} H_l(\pi) + b(\pi'(\pi))R(\pi) - g, \\ b_{PM(m)}(\pi) = C'_{PM(m)}, m = 1, \cdots, M-1 \\ b_{OB}(\pi) = C_{OB} + \sum_{i=1}^{M} b(e_i)\pi_i \end{cases} \tag{4.18}$$

IV.3. Existing solution method - pure recursive technique

First, let us consider a sample path emanating from an information state π. By a sample path, we mean the sequence of information states over time when no action is taken, which is denoted by $\{\pi, \pi^2, \cdots, \Pi(\pi)\}$ where $\pi^2 = \pi'(\pi)$, $\pi^3 = \pi'(\pi^2)$ and so on. $\Pi(\pi)$, defined by $\Pi(\pi) \equiv \pi^{k^*}$, where $k^* = \min\{k : ||\pi^{k+1} - \pi^k|| < \epsilon\}$ with small $\epsilon > 0$, is a stationary state or an absorbing state. Maillart (2006) shows, by referring to Mandl (1959), that when the Markov chain is acyclic, $\Pi(\pi)$ exists for any $\epsilon > 0$.

Let us call the sequence of states emanating from one of the extreme points $b(e_i)$, $\forall i$, in (4.18) an *extreme* sample path. Since all the biases at the states on the extreme sample paths are independent of the biases at the states on non-extreme sample paths, we can easily obtain $b(e_i)$, $\forall i$ and average cost g by applying *policy*

iteration (or *value iteration*) methods to the states only on the extreme sample paths (Puterman, 1994). Then, $b_{OB}(\pi)$ and $b_{PM}(\pi)$ in (4.18) can be directly computed.

Now, we only need to compute $b_{NA}(\pi)$ to get $b(\pi)$. We can use the similar recursive technique introduced by Maillart (2004). First, we solve (4.18) for $\Pi(\pi)$ by

$$b(\Pi(\pi)) = min \begin{cases} b_{NA}(\Pi(\pi)) = \frac{\sum_{l=1}^{L} C'_{CM} H_l(\pi) - g}{1 - R(\Pi(\pi))}, \\ b_{PM}(\Pi(\pi)) = C'_{PM}, \\ b_{OB}(\Pi(\pi)) = C_{OB} + \sum_{i=1}^{M} b(e_i)\Pi(\pi)_i \end{cases} \tag{4.19}$$

Then, we apply $b(\Pi(\pi))$ to (4.18) in order to find the optimal policy at the previous state. By solving the recursive set of equations backwards, we can get the optimal policy along the states on the sample path emanating from the original state π.

However, this recursive technique might be computationally inefficient when we want to find the optimal policies at a large number of states in a high dimensional state space. This is because we have to apply the recursive set of equations for each state. These computational difficulties motivate us to study the structural properties of the model.

IV.4. Structural properties

In this section we characterize the optimal policy of the static CBM model with several structural properties. More specifically, we derive a set of closed expressions for the optimal policy including the exact control limits for PM. In later sections we show how these results help attain optimal polices. We also show that the model exhibits the monotonous AM4R policy structure. This finding is an extension of a previous study in Maillart (2006). In Maillart (2006), the AM4R results are shown for a simpler model than the one presented here, and are obtained under specific assumptions on the transition matrix and information states. We relax the assumptions while proving

the results, and establish the conditions when the optimal policy is simplified to a more intuitive "At-Most-Three-Region" (AM3R) structure.

For simplicity, we consider one failure mode and one major PM level, $PM(1)$ in the following discussions in this chapter (But the results will be extended for general cases with multiple failure modes and multiple repair levels in Chapter VII). We use PM to denote $PM(1)$, and use W_{CM} and W_{PM} to represent the harsh weather probabilities which prohibit CM and PM, respectively. Similarly, C_{CM} and C_{PM} represent the direct repair costs for CM and PM, respectively. Typically CM requires more complicated repair jobs than PM, resulting in $W_{PM} \leq W_{CM}$ and $C_{PM} \leq C_{CM}$ in many practical cases.

IV.4.1. Preliminary results

We first introduce several definitions which are often used in POMDP studies. These definitions can be found, for example, in Lovejoy (1987), Rosenfield (1976), and Ohnishi *et al.* (1986).

Definition 1. *Information state π is stochastically less (or smaller) than $\hat{\pi}$, denoted as $\pi \prec_{st} \hat{\pi}$ if and only if $\sum_{i \geq j} \pi_i \leq \sum_{i \geq j} \hat{\pi}_i$ for all j.*

Definition 2. *Information state π is less (or smaller) in likelihood ratio than $\hat{\pi}$, denoted as $\pi \prec_{lr} \hat{\pi}$ if and only if $\pi_i \hat{\pi}_j - \pi_j \hat{\pi}_i \geq 0$ for all $j \geq i$.*

These two definitions present the binary relations of the two states in the sense of deterioration. Both definitions imply that when the system is less deteriorated, the state is stochastically (or in the likelihood ratio) less than another (Maillart and Zheltova, 2007). However, as Proposition 1(a) (see below) suggests, $\prec_{lr}$ relationship is stronger than $\prec_{st}$ relationship (Rosenfield, 1976). We also need additional definitions regarding the transition matrix P.

Definition 3. *A transition matrix P has an Increasing Failure Rate (IFR) if $\sum_{j\geq k} p_{ij} \leq \sum_{j\geq k} p_{i'j}$ for all $i' \geq i$ and all k.*

Definition 4. *A transition matrix P is Totally Positive of order 2 (TP2) if $p_{ij}p_{i'j'} \geq p_{i'j}p_{ij'}$ for all $i' \geq i$ and $j' \geq j$.*

These definitions imply that the more deteriorated system tends to more likely deteriorate further and/or fail (Maillart, 2006). Similar to the stochastic relations defined in Definition 1 and Definition 2, $TP2$ is more stringent assumption than IFR due to the following Proposition 1(b) (Rosenfield, 1976).

Proposition 1. *(Rosenfield, 1976) (a) If $\pi \prec_{lr} \hat{\pi}$, then $\pi \prec_{st} \hat{\pi}$. (b) If P is $TP2$, then P is IFR.*

Before presenting our results, we introduce several well-known results in the following two Propositions.

Proposition 2. *(Derman, 1963) For any column vector v such that $v_i \leq v_{i+1}, \forall i$, if $\pi \prec_{st} \hat{\pi}$, then $\pi \cdot v \leq \hat{\pi} v$.*

Proposition 3. *(a) (Maillart, 2006) Suppose that P is IFR. If $\pi \prec_{st} \hat{\pi}$, then $R(\pi) \geq R(\hat{\pi})$. (b) (Maillart and Zheltova, 2007) If P is IFR and $\pi \prec_{st} \hat{\pi}$, then $\pi P \prec_{st} \hat{\pi} P$*

The next Proposition 4 establishes that when P is IFR, the stochastic ordering of two states are maintained after the transitions.

Proposition 4. *Suppose that P is IFR. If $\pi \prec_{st} \hat{\pi}$, $\pi'(\pi) \prec_{st} \pi'(\hat{\pi})$.*

Proof Let $(\pi P)_i$, $\pi'_i(\pi)$ and $\pi'_i(\hat{\pi})$ denote the ith position of the row vector πP, $\pi'(\pi)$ and $\pi'(\hat{\pi})$, respectively. Then, we have

$$\sum_{i\geq j} \pi'_i(\pi) \;=\; \sum_{i\geq j} \frac{(\pi P)_i}{R(\pi)} \leq \sum_{i\geq j} \frac{(\pi P)_i}{R(\hat{\pi})} \leq \sum_{i\geq j} \frac{(\hat{\pi} P)_i}{R(\hat{\pi})} = \sum_{i\geq j} \pi'_i(\hat{\pi}) \qquad (4.20)$$

The two inequalities in (4.20) hold due to Proposition 3(a) and Proposition 3(b), respectively. □

The following Proposition 5 demonstrates that the total cost-to-go for a failed system is always greater than, or equal to, the cost-to-go when PM is selected.

Proposition 5. *(a)* $CM_n(e_{M+1}) - C_{CM} \geq PM_n - C_{PM}$ *for* $\forall n$ *where* $CM_n(e_{M+1})$, *and* PM_n *are defined in (4.2), and (4.3), respectively. (b)* $CM_n(e_{M+1}) \geq PM_n$ *for* $\forall n$.

Proof (a) We prove the claim by induction method. Let λ is the lead time to prepare repair resources upon failure (i.e. $\lambda = \lambda(1)$). Suppose that $n \geq \lambda + 1$ because one cannot carry out corrective maintenance when the system fails and the number of remaining periods is less than, or equal to, the lead time. Without loss of generality, we suppose $V_{\lambda+1}(\pi) = 0$ for an operating system and $CM_{\lambda+1}(e_{M+1}) = C_{CM}$ and $PM_{\lambda+1} = C_{PM}$. Then, $CM_{\lambda+1}(e_{M+1}) - C_{CM} = PM_{\lambda+1} - C_{PM} = \tau$. Suppose that $CM_n(e_{M+1}) - C_{CM} \geq PM_n - C_{PM}$ for $n \geq \lambda + 1$. Then,

$$
\begin{aligned}
& CM_{n+1}(e_{M+1}) - C_{CM} \\
& = (1 - W_{CM})(\tau + C_{CM} + V_n(e_1)) + W_{CM}(\tau + CM_n(e_{M+1})) - C_{CM} && (4.21) \\
& = \tau + V_n(e_1) + W_{CM}(CM_n(e_{M+1}) - C_{CM} - V_n(e_1)) && (4.22) \\
& \geq \tau + V_n(e_1) + W_{PM}(PM_n - C_{PM} - V_n(e_1)) && (4.23) \\
& = PM_{n+1} - C_{PM}, && (4.24)
\end{aligned}
$$

where (4.23) is from induction hypothesis. Therefore, $CM_n(e_{M+1}) - C_{CM} \geq PM_n - C_{PM}$ holds for $\forall n \geq \lambda + 1$.

(b)

$$CM_n(e_{M+1}) = (1 - W_{CM})(\tau + C_{CM} + V_{n-1}(e_1)) + W_{CM}(\tau + CM_{n-1}(e_{M+1})), \tag{4.25}$$

$$= \tau + C_{CM} + V_{n-1}(e_1) + W_{CM}(CM_{n-1}(e_{M+1}) - C_{CM} - V_{n-1}(e_1)) \tag{4.26}$$

$$\geq \tau + C_{PM} + V_{n-1}(e_1) + W_{PM}(PM_{n-1} - C_{PM} - V_{n-1}(e_1)) \tag{4.27}$$

$$= PM_n. \tag{4.28}$$

Inequality in (4.27) is due to the result of Proposition 5(a) and the fact that $C_{CM} \geq C_{PM}$ and $W_{CM} \geq W_{PM}$. Consequently, $CM_n(e_{M+1}) \geq PM_n$ for all n. □

The above Propositions allow us to derive the monotonicity of $V_n(\pi)$ in $\prec_{st}$-ordering, as shown in Lemma 1.

Lemma 1. *If P is IFR, $b(\pi)$ in (4.18) is non-decreasing in $\prec_{st}$-ordering.*

Proof By induction, we can show that $V_n(\pi)$ is non-decreasing in $\prec_{st}$ when P is IFR. Let λ is the lead time to prepare repair resources upon failure. Suppose that $n \geq \lambda+1$. Without loss of generality, we suppose that $V_{\lambda+1}(\pi) = 0$, $\forall\pi$. Then, $NA_{\lambda+2}(\pi) = (\tau\lambda + CM_{\lambda+1}(e_{M+1}))(1 - R(\pi))$ is non-decreasing in $\prec_{st}$ from Proposition 3(a), and $PM_{\lambda+2}$ is constant in π. $OB_{\lambda+2}(\pi) = C_{OB} + \sum_{i=1}^{M} min\{NA_{\lambda+2}(e_i), PM_{\lambda+2}\}\pi_i$. Since $e_i \prec_{st} e_j$ for $i \leq j$ and $NA_{\lambda+2}(e_i)$ is nondecreasing in i, $OB_{\lambda+2}(\pi)$ is also non-decreasing in $\prec_{st}$ due to Proposition 2. Therefore, $V_{\lambda+2}(\pi)$ is non-decreasing in $\prec_{st}$.

Suppose that $V_n(\pi)$ is non-decreasing in $\prec_{st}$ for $\forall n \geq \lambda + 1$. Then, for $\pi \prec \hat{\pi}$,

$$NA_{n+1}(\pi) = (\tau\lambda + CM_{n-\lambda}(e_{M+1}))(1 - R(\pi)) + V_n(\pi^2)R(\pi) \tag{4.29}$$

$$\leq (\tau\lambda + CM_{n-\lambda}(e_{M+1}))(1 - R(\pi)) + V_n(\hat{\pi}^2)R(\pi) \tag{4.30}$$

$$= (\tau\lambda + CM_{n-\lambda}(e_{M+1})) - (\tau\lambda + CM_{n-\lambda}(e_{M+1}) - V_n(\hat{\pi}^2))R(\pi) \tag{4.31}$$

$$\leq (\tau\lambda + CM_{n-\lambda}(e_{M+1})) - (\tau\lambda + CM_{n-\lambda}(e_{M+1})) - V_n(\hat{\pi}^2))R(\hat{\pi}) \tag{4.32}$$

$$= (\tau\lambda + CM_{n-\lambda}(e_{M+1}))(1 - R(\hat{\pi})) + V_n(\hat{\pi}^2)R(\hat{\pi})) = NA_{n+1}(\hat{\pi}) \tag{4.33}$$

Note that $H_1(\pi) = 1 - R(\pi)$ for $L = 1$ in (4.29). (4.30) follows from the induction assumption and Proposition 4(a). (4.32) follows from Proposition 3(a) and the fact that $\tau\lambda + CM_{n-\lambda}(e_{M+1}) \geq V_n(\pi), \forall\pi$ (Note that $\tau\lambda + CM_{n-\lambda}(e_{M+1})$ is the revenue losses during the lead time *plus* corrective maintenance costs when the system fails, so it is always greater than the optimal value function for any operating state). It is obvious that $OB_{n+1}(\pi) = C_{OB} + \sum_i min\{NA_{n+1}(e_i), PM_{n+1}\}\pi_i$ is also non-decreasing in $\prec_{st}$ with the similar reason explained above. Consequently, $V_{n+1}(\pi)$ is nondecreasing in $\prec_{st}$, $\forall n \geq \lambda + 1$. Since $b(\pi)$ can be obtained by taking limits of $V_n(\pi)$, $b(\pi)$ is nondecreasing in $\prec_{st}$, which concludes the claim. □

The claim of Lemma 1 extends the result presented in Maillart (2006) where the monotonicity of the optimal cost function in $\prec_{lr}$-ordering on the $TP2$ transition matrix is shown. Also, unlike our model, the model in Maillart (2006) does not consider revenue losses, lead time and stochastic operating environments (i.e. $\tau = 0$, $\lambda = 0$, $W_{CM} = W_{PM} = 0$). Therefore, the result of Lemma 1 is more general, and can be applied to other general aging systems.

IV.4.2. Closed expressions for optimal policy regions

In this section, we present the closed boundary expressions for the optimal policy. Let $\Omega_{NA}(\pi), \Omega_{OB}(\pi), \Omega_{PM}(\pi)$ be the set of information states with $\delta^S(\pi) = NA$, $\delta^S(\pi) = OB$, and $\delta^S(\pi) = PM$, respectively. To get the optimal policy to minimize the long-run average cost, we need to compare $b_{NA}(\pi)$, $b_{PM}(\pi)$, and $b_{OB}(\pi)$.

First, the following Lemma 2 explains when NA is preferred to PM, and vice versa. To prove the claim, we apply a technique similar to the one used in Ghasemi *et al.* (2007).

Lemma 2. *Suppose that P is IFR and upper-triangular. $\delta^S(\pi) \neq PM$ if $R(\pi) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$. Also, $\delta^S(\pi) \neq NA$ if $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$ for $\pi \prec_{st} \pi'(\pi)$.*

Proof

$$b_{NA}(\pi) - b_{PM}(\pi) = C'_{CM}(1 - R(\pi)) + b(\pi'(\pi))R(\pi) - g - C'_{PM} \tag{4.34}$$

$$= (C'_{CM} - C'_{PM})(1 - R(\pi)) - g + (b(\pi'(\pi)) - C'_{PM})R(\pi) \tag{4.35}$$

Note that $b(\pi'(\pi)) \leq C'_{PM}$. Consequently, if $(C'_{CM} - C'_{PM})(1 - R(\pi)) - g \leq 0$ (or equivalently, $R(\pi) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$), NA is preferred to PM. Next, consider the case that $(C'_{CM} - C'_{PM})(1 - R(\pi)) - g > 0$. Let us assume that $\delta^S(\pi) = NA$. Then,

$$b(\pi'(\pi)) - b(\pi) = b(\pi'(\pi)) - (C'_{CM}(1 - R(\pi)) + b(\pi'(\pi))R(\pi) - g) \tag{4.36}$$

$$= (b(\pi'(\pi)) - C'_{PM})(1 - R(\pi)) - (C'_{CM} - C'_{PM})(1 - R(\pi)) + g \tag{4.37}$$

(4.36) holds from the assumption $\delta^S(\pi) = NA$ and thus, $b(\pi) = C'_{CM}(1 - R(\pi)) + b(\pi'(\pi))R(\pi) - g$. Note that in (4.37), $b(\pi'(\pi)) \leq C'_{PM}$. Therefore, when $(C'_{CM} - C'_{PM})(1 - R(\pi)) - g > 0$, $b(\pi'(\pi)) \leq b(\pi)$ with the assumption of $\delta^S(\pi) = NA$. But, this result contradicts that $b(\pi'(\pi)) \geq b(\pi)$ for $\pi \prec_{st} \pi'(\pi)$ from Lemma 1. Therefore,

when $(C'_{CM} - C'_{PM})(1 - R(\pi)) - g > 0$, or equivalently, $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$, NA cannot be optimal. □

The claim of Lemma 2 is intuitive. As the system deteriorates, its reliability monotonically decreases. When its reliability is lower than a threshold (here, it is $1 - \frac{g}{C'_{CM} - C'_{PM}}$), it is better to take some actions rather than do nothing. On the contrary, we need not carry out costly maintenance action for a highly reliable system. Note that the second part of Lemma 2 requires the assumption $\pi \prec_{st} \pi'(\pi)$, which implies that the next state is more deteriorated than the current state in a probabilistic sense. This assumption should hold in most commonly encountered aging systems.

With the result of Lemma 2, $b_{OB}(\pi)$ in (4.19) can be reformulated as follows:

$$b_{OB}(\pi) = C_{OB} + \sum_{i=1}^{M} \left\{ b_{NA}(e_i) \cdot 1\left(R(e_i) \geq \alpha\right) + b_{PM}(e_i) \cdot 1\left(R(e_i) < \alpha\right) \right\} \pi_i \quad (4.38)$$

Here, $\alpha = 1 - \frac{g}{C'_{CM} - C'_{PM}}$. $OB_n(\pi)$ in (4.6) can be reformulated likewise.

Next, let us compare $b_{OB}(\pi)$ with $b_{PM}(\pi)$. If $C'_{PM} < C_{OB} + \sum_i b(e_i)\pi_i$, PM is preferred to OB. As a result, if $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$, and $C'_{PM} < C_{OB} + \sum_i b(e_i)\pi_i$, the optimal policy is PM. Also, from the facts that $b_{OB}(\pi)$ is non-decreasing in $\prec_{st}$-ordering, and that $b_{PM}(\pi)$ is constant, we can derive the control limit for PM in a closed form. Many previous maintenance studies based on POMDPs simply prove the "existence" of the control limit for PM. But for this problem, we analytically obtain the necessary and sufficient condition. Theorem IV.1 summarizes the results.

Theorem IV.1. *Suppose that P is IFR. (a) For $\pi \prec_{st} \pi'(\pi)$, the region where the optimal policy is PM is defined by $\Omega_{PM} = \{\pi; R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}, C'_{PM} < C_{OB} + \sum b(e_i)\pi_i\}$, whereas PM cannot be optimal for $\pi \notin \Omega_{PM}$. (b) Furthermore, if $\delta^S(\pi) =$*

PM, $\delta^S(\hat{\pi}) = PM$ for $\pi \prec_{st} \hat{\pi}$.

Proof The first part is straightforward from Lemma 2 and the above discussions. Regarding the second part, NA cannot be optimal at $\hat{\pi}$ from the fact that $R(\hat{\pi}) \leq R(\pi)$ for $\pi \prec_{st} \hat{\pi}$. Also, since $b(e_i)$ is non-decreasing in i, $\sum_i b(e_i)\pi_i$ is also non-decreasing in $\prec_{st}$-ordering from Proposition 2, and so is $b_{OB}(\pi)$. This leads to $b_{OB}(\hat{\pi}) \geq b_{OB}(\pi)$. But, $b_{PM}(\pi)$ is constant. Thus, when $\delta^S(\pi) = PM$, OB cannot be optimal at $\hat{\pi}$ as well, which concludes the second part of the Theorem. □

This PM region in Theorem IV.1 defines the optimal PM region of the AM4R policy, as we will discuss in Section IV.4.4.

Corollary 1. *Suppose that P is IFR. (a) If $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$, and $C'_{PM} \geq C_{OB} + \sum b(e_i)\pi_i$, $\delta^S(\pi) = OB$ for $\pi \prec_{st} \pi'(\pi)$. (b) If $R(\pi) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$, and $C'_{PM} < C_{OB} + \sum b(e_i)\pi_i$, $\delta^S(\pi) = NA$.*

Proof It follows directly from Lemma 2 and the fact that OB is preferred to PM when $C'_{PM} \geq C_{OB} + \sum b(e_i)\pi_i$. □

Finally, let us compare $b_{NA}(\pi)$ with $b_{OB}(\pi)$. We present the conditions under which NA is preferred to OB and vice versa, in Lemma 3 and Lemma 4.

Lemma 3. *Suppose that $C_{OB} + C_{PM} \leq C_{CM}$. If $R(\pi) \geq \frac{(C'_{CM} - C_{OB} - \sum b(e_i)\pi_i) - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}$, then $\delta^S(\pi) \neq OB$.*

Proof We use similar technique used in Lemma 2.

$$b_{NA}(\pi) - b_{OB}(\pi) \tag{4.39}$$

$$= C'_{CM}(1 - R(\pi)) + b(\pi'(\pi))R(\pi) - g - C_{OB} - \sum b(e_i)\pi_i \tag{4.40}$$

$$= (C'_{CM} - C_{OB} - \sum b(e_i)\pi_i)(1 - R(\pi)) - g + R(\pi)(b(\pi'(\pi)) - C_{OB} - \sum b(e_i)\pi_i) \tag{4.41}$$

$$= (C'_{CM} - C_{OB} - \sum b(e_i)\pi_i)(1 - R(\pi)) - g + R(\pi) \sum b(e_i)(\pi'_i(\pi) - \pi_i)$$
$$+ R(\pi)(b(\pi'(\pi)) - C_{OB} - \sum b(e_i)\pi'_i(\pi)), \tag{4.42}$$

Note that $b(\pi'(\pi)) \leq C_{OB} + \sum b(e_i)\pi'_i(\pi)$. Therefore, if $(C'_{CM} - C_{OB} - \sum b(e_i)\pi_i)(1 - R(\pi)) - g + R(\pi) \sum b(e_i)(\pi'_i(\pi) - \pi_i) \leq 0$, $b_{NA}(\pi) \leq b_{OB}(\pi)$. Re-arranging the condition yields

$$(C'_{CM} - C_{OB} - \sum b(e_i)\pi_i)(1 - R(\pi)) - g + R(\pi) \sum b(e_i)(\pi'_i(\pi) - \pi_i) < 0 \tag{4.43}$$

$$\Leftrightarrow R(\pi) \geq \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)} \tag{4.44}$$

The last inequality (4.44) comes from $b(e_i) \leq C'_{PM}$ for all $i = 1, \cdots, M$ and from $C_{OB} + C'_{PM} \leq C'_{CM}$ (Note that $C_{OB} + C_{PM} \leq C_{CM}$ by assumption). □

Similar to Lemma 2, Lemma 3 also explains that when the system is in a fairly good condition with a high reliability, we need not carry out costly inspection of the system. Along with Lemma 2, the following Corollary 2 specifies the sufficient condition for NA to be optimal.

Corollary 2. *If* $R(\pi) \geq \max\{1 - \frac{g}{C'_{CM} - C'_{PM}}, \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}\}$, *then* $\delta^S(\pi) = NA$.

Proof It follows directly from Lemma 2 and Lemma 3. □

Lemma 4 specifies the sufficient condition under which OB is optimal.

Lemma 4. *Suppose that* $R(\pi) < \frac{C'_{CM}-C_{OB}-\sum b(e_i)\pi_i-g}{C'_{CM}-C_{OB}-\sum b(e_i)\pi'_i(\pi)}$. *If* $\delta^S(\pi'(\pi)) = OB$, *then* $\delta^S(\pi) = OB$.

Proof We will use contradiction. Assume that $\delta^S(\pi) = NA$. Then,

$$b(\pi'(\pi)) - b(\pi) - \sum b(e_i)(\pi'_i(\pi) - \pi_i) \tag{4.45}$$

$$= b(\pi'(\pi)) - C'_{CM}(1 - R(\pi)) - b(\pi'(\pi))R(\pi) + g - \sum b(e_i)(\pi'_i(\pi) - \pi_i) \tag{4.46}$$

$$= (b(\pi'(\pi)) - C'_{CM})(1 - R(\pi)) + g - \sum b(e_i)(\pi'_i(\pi) - \pi_i) \tag{4.47}$$

$$= (b(\pi'(\pi)) - C_{OB} - \sum b(e_i)\pi'_i(\pi))(1 - R(\pi)) +$$
$$(C_{OB} + \sum b(e_i)\pi_i - C'_{CM})(1 - R(\pi)) + g - R(\pi)\sum b(e_i)(\pi'_i(\pi) - \pi_i) \tag{4.48}$$

Note that $b(\pi'(\pi)) - C_{OB} - \sum b(e_i)\pi'_i(\pi) \leq 0$. Also, by the condition of the claim, the remaining term is also negative. Therefore, we get $b(\pi'(\pi)) - b(\pi) - \sum b(e_i)(\pi'_i(\pi) - \pi_i) < 0$ under the assumption of $\delta^S(\pi) = NA$. However,

$$b(\pi'(\pi)) - b(\pi) - \sum b(e_i)(\pi'_i(\pi) - \pi_i) \tag{4.49}$$

$$= b_{OB}(\pi'(\pi)) - b(\pi) - b_{OB}(\pi'(\pi)) + b_{OB}(\pi) \quad (\text{from } \delta^S(\pi'(\pi)) = OB), \tag{4.50}$$

$$= -b(\pi) + b_{OB}(\pi) \geq 0, \tag{4.51}$$

which contradicts the assumption. As a result, $\delta^S(\pi)$ cannot be NA. Also note that $b_{OB}(\pi) \leq b_{OB}(\pi'(\pi)) \leq b_{PM}(\pi'(\pi)) = b_{PM}(\pi)$. Therefore, PM cannot be also optimal, which concludes $\delta^S(\pi) = OB$. □

Table 1 summarizes the closed boundaries for the optimal policy of the static CBM model with one failure mode and one PM level.

IV.4.3. Structural properties along sample path

By extending the claim of Proposition 4(a), we can show that when P is IFR, all of the states in the sample path emanating from any π is in increasing stochastic

Table 1. Closed boundaries for optimal policy when $L = 1$ and $M = 1$

$\delta^S(\pi)$	Conditions	Remark
PM	if and only if $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$ and	Sufficient
	$C'_{PM} < C_{OB} + \sum b(e_i)\pi_i$ for $\pi \prec_{st} \pi'(\pi)$	and necessary
OB	if $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$ and $C'_{PM} \geq C_{OB} + \sum b(e_i)\pi_i$	Sufficient
	or, if $R(\pi) \leq \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}$ and $\delta^S(\pi'(\pi)) = OB$	
NA	if $R(\pi) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$ and $C'_{PM} < C_{OB} + \sum b(e_i)\pi_i$	Sufficient
	or, if $R(\pi) \geq \max\{1 - \frac{g}{C'_{CM} - C'_{PM}}, \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}\}$	

order as long as $\pi \prec_{st} \pi'(\pi)$. This allows us to apply all of the results developed in Section IV.4.2 to the states along a sample path in increasing stochastic order. The following Corollary 3 summarizes them.

Corollary 3. *Suppose that P is IFR. Then the states along a sample path satisfy the following properties for $\pi \prec_{st} \pi'(\pi)$.*

(a) Any sample path is in $\prec_{st}$-increasing order. i.e. $\pi \prec_{st} \pi^2 (= \pi'(\pi)) \prec_{st} \pi^3 \prec_{st} , \cdots , \prec_{st} \Pi(\pi)$ where $\pi^{k+1} = \pi'(\pi^k)$.

(b) $V_n(\pi)$ and $b(\pi)$ are non-decreasing along any sample path.

(c) Suppose that $R(\pi^q) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$. $\delta^S(\pi^k) \neq PM$ for $\forall k \leq q$. On the contrary, if $R(\pi^q) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$, $\delta^S(\pi^k) \neq NA$ for $\forall k \geq q$.

(d) There exists a critical number k^* such that $\delta^S(\pi^k) = PM$, $\forall k \geq k^*$, and $\delta^S(\pi^k) \neq$

PM otherwise. And, such k^* *is given by* $k^* = \max\{k1(\pi), k2(\pi)\}$ *where*

$$k1(\pi) = min\{k; R(\pi^k) < 1 - \frac{g}{C'_{CM} - C'_{PM}}\} \quad (4.52)$$

$$k2(\pi) = min\{k; C_{OB} + \sum b(e_i)\pi_i^k > C'_{PM}\}. \quad (4.53)$$

Proof (a) Applying Proposition 4 repeatedly to both sides of this inequality yields the result.

(b) Since the states along any sample path is in $\prec_{st}$-increasing order, the result follows directly from Lemma 1.

(c) Note that $R(\pi^k)$ is non-increasing in k by proposition 3(a). Then, the result follows from Lemma 2.

(d) For $k \geq k1(\pi)$, NA cannot be the optimal action from Lemma 2. Also, for $k \geq k2(\pi)$, PM is preferable to OB since $C_{OB} + \sum b(e_i)\pi_i^k$ is nondecreasing in k in a $\prec_{st}$-increasing sample path and C'_{PM} is constant. Hence for $k \geq k^*$, either NA or OB cannot be optimal. For $k1 \leq k < k^*$, OB is optimal, whereas $k2 \leq k < k^*$, NA is optimal. For $k < min\{k1, k2\}$, OB or NA is optimal. □

IV.4.4. The monotonic policy

Several previous studies establish the AM4R policy structure along an ordered subset of state space for POMDP problems in different maintenance settings. For example, Maillart (2006) presents the AM4R structure along any straight line of $\prec_{lr}$-ordered information states when P is $TP2$ in her model.

In this section, we establish similar results for the presented problem under less stringent assumptions on the transition matrix and information states. We also deal with more complicated problem which reflect the special characteristics of wind farm operations. Specifically, we show that the optimal policy has the AM4R structure

along a straight line of $\prec_{st}$-ordered states on IFR transition matrix. In deriving the desired results, we take an analogous approach to one proposed in Maillart (2004).

Consider two states π and $\hat{\pi}$ for $\pi \prec_{st} \hat{\pi}$. Let us denote a state between π and $\hat{\pi}$ by $\pi(\eta) = \eta\pi + (1-\eta)\hat{\pi}$, $0 \leq \eta \leq 1$. Here, higher η implies a more deteriorated condition (we will show the reason in the proof of Theorem IV.2). Then, there exist at most three numbers η_1, η_2, η_3 to divide the optimal policy regions as follows:

$$\delta^S(\pi(\eta)) = min \begin{cases} NA, & \text{if } \eta < \eta_1 \text{ or } \eta_2 < \eta \leq \eta_3 \\ OB, & \text{if } \eta_1 \leq \eta \leq \eta_2 \\ PM, & \text{if } \eta > \eta_3 \end{cases} \tag{4.54}$$

That is, as η increases, the optimal policy regions are divided into at most four regions with the order $NA \rightarrow OB \rightarrow NA \rightarrow PM$. To establish this AM4R structure, we first show the concavity of $V_n(\pi)$.

Lemma 5. *$V_n(\pi)$ is piecewise linear concave for all n.*

Proof We apply the similar induction technique used in Maillart (2006). Let λ denote the lead time. Suppose that $n \geq \lambda + 1$. Without loss of generality, we assume $V_{\lambda+1}(\pi) = 0$ for an operating system. $NA_{\lambda+1}(\pi) = \lambda\tau(1 - R(\pi))$ is linear in π. $OB_n(\pi)$ is hyperplane of π and PM_n is constant in π for $\forall n$. Therefore, $V_{\lambda+1}(\pi)$ is piecewise linear concave because minimum of linear functions is piecewise linear concave. Now, suppose that $V_n(\pi)$, $n \geq \lambda + 1$, is piecewise linear concave such that $V_n(\pi) = min\{\pi \cdot a_n^T; a_n \in A_n\}$ where a_n is a $1 \times (M+1)$ dimensional column vector. We only need to examine $NA_{n+1}(\pi)$ to show the piecewise linear concavity of $V_{n+1}(\pi)$. The first term of $NA_{n+1}(\pi)$, i.e. $\lambda\tau + CM_{n-\lambda-1}(e_M + 1))(1 - R(\pi))$, is linear in π.

The second term of $NA_{n+1}(\pi)$ is,

$$R(\pi)V_n(\pi^2) = R(\pi)min\{\pi^2 \cdot a_n^T; a_n \in A_n\} \tag{4.55}$$

$$= R(\pi)min\left\{\left[\frac{(\pi P)_1}{R(\pi)}, \frac{(\pi P)_2}{R(\pi)}, \cdots, \frac{(\pi P)_M}{R(\pi)}, 0\right] \cdot a_n^T; a_n \in A_n\right\} \tag{4.56}$$

$$= min\{[(\pi P)_1, (\pi P)_2, \cdots, (\pi P)_M, 0] \cdot a_n^T; a_n \in A_n\} \tag{4.57}$$

$$= min\{\pi \cdot a_{n+1}^T; a_{n+1} \in A_{n+1}\} \tag{4.58}$$

Since $R(\pi)V_n(\pi^2)$ is the minimum of hyperplanes, it is piecewise linear concave, which makes $NA_{n+1}(\pi)$ is also piecewise linear concave. Consequently, $V_{n+1}(\pi)$ is piecewise linear concave. And the claim holds for $\forall n \geq \lambda + 1$ by induction. □

Now, we are ready to prove the monotonic AM4R structure along a $\prec_{st}$-increasing line.

Theorem IV.2. *If P is IFR, the optimal policy has the monotonic AM4R structure along any straight line of information states in $\prec_{st}$-increasing order. Furthermore, the control limit to define optimal PM policy is defined by*
$\eta^* = \inf\{\lambda; R(\pi(\lambda)) < 1 - \frac{g}{C'_{CM} - C'_{PM}}, C'_{PM} < C_{OB} + \sum b(e_i)\pi(\lambda)_i\}$.

Proof Consider the two states $\pi(\eta_1)$ and $\pi(\eta_2)$ between π and $\hat{\pi}$ ($\pi \prec_{st} \hat{\pi}$) where $\pi(\eta_j) = \eta_j\pi + (1-\eta_j)\hat{\pi}$, for $j = 1, 2$ and $0 \leq \eta_1 \leq \eta_2 \leq 1$. Then, from $\sum_{i \geq j} \pi_i \prec_{st} \eta_1 \sum_{i \geq j} \pi_i + (1-\eta_1) \sum_{i \geq j} \hat{\pi}_i \prec_{st} \sum_{i \geq j} \hat{\pi}_i$, we have $\pi \prec_{st} \pi(\eta_1) \prec_{st} \hat{\pi}$. In a similar way, we can easily show that $\pi(\eta_1) \prec_{st} \pi(\eta_2) \prec_{st} \hat{\pi}$. Therefore, $\pi(\eta)$ is in $\prec_{st}$-increasing in η, which implies that $b_{NA}(\pi(\eta))$ and $b_{OB}(\pi(\eta))$ is non-decreasing in η. $b_{PM}(\pi(\eta))$ is constant. Hence, there exists a control limit η^* such that for any $\eta > \eta^*$, PM is optimal. The value of η^* is straightforward from Theorem IV.1. Next, let us consider $0 \leq \eta \leq \eta^*$. For this region, we already know that PM cannot be optimal from Theorem IV.1. In Lemma 5, we show that $NA_n(\pi)$ is piecewise linear concave.

Thus $b_{NA}(\pi)$ is also piecewise linear concave, but $b_{OB}(\pi)$ is hyperplane. Thus, $\{\pi; b_{NA}(\pi) \geq b_{OB}(\pi)\}$ is a convex set and $\{\eta; b_{NA}(\pi(\eta)) \geq b_{OB}(\pi(\eta)), 0 \leq \eta \leq \eta^*\}$ is also a convex set. This concludes the AM4R structure. □

As Rosenfield (1976) points out, the second NA region in the AM4R structure may seem counter-intuitive. In the following discussions, we establish the conditions under which we have the more intuitive AM3R policy structure. Let us define the critical numbers to divide the optimal policy regions as follows:

$$\eta_{NA \leq PM} = \max\{\eta; R(\pi(\lambda)) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}\} \tag{4.59}$$

$$\eta_{OB \leq PM} = \max\{\eta; C_{OB} + \sum b(e_i)\pi_i(\eta) \leq C'_{PM}\}, \tag{4.60}$$

where $\pi_i(\eta)$ in (4.60) is i^{th} element of $\pi(\eta)$. Note that for $\eta \leq \eta_{NA \leq PM}$, NA is preferred to PM and vice versa. Similarly, For $\eta \leq \eta_{OB \leq PM}$, OB is preferred to PM and vice versa.

Corollary 4. *If $\eta_{NA \leq PM} < \eta_{OB \leq PM}$, the optimal policy has the monotonic AM3R structure along any $\prec_{st}$-increasing straight line of information states with the order of $NA \rightarrow OB \rightarrow PM$. The optimal policy region for PM is given by $\{\pi(\eta); C'_{PM} < C_{OB} + \sum b(e_i)\pi_i(\eta)\}$.*

Proof When $\eta_{NA \leq PM} < \eta_{OB \leq PM}$, The second NA region of AM4R structure vanishes. So the optimal policy structure results in at most three regions. The optimal policy region for PM is straightforward from the previous discussions. □

Fig. 6 compares the two policy structures. Whether the optimal policy structure exhibits the AM4R or AM3R is highly dependent on the costs of PM. When the PM costs are relatively larger compared to the costs related to other actions, the structure

is more likely to result in the AM4R structure, as shown in Fig. 6(a). Otherwise, when PM costs are comparable to other costs, the AM3R structure occurs more likely, as shown in Fig. 6(b).

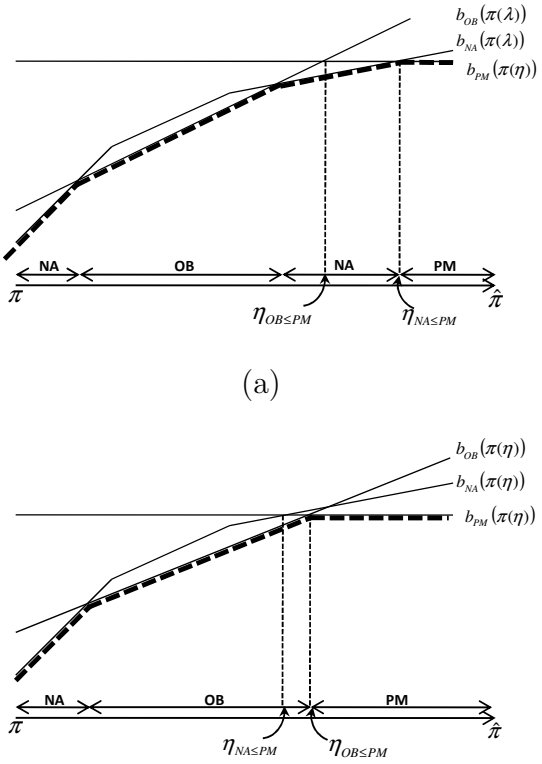


Fig. 6. Monotonic optimal policy structure: (a) AM4R structure (b) AM3R structure

In wind turbine operations, the repair costs (C'_{CM}) after an unplanned failure are considerably larger compared to the PM costs. Also, in most cases OB costs are not negligible because inspecting the physical condition by dispatching crew is costly due to the high labor costs and the long distance of wind farms from the operation centers (Rademakers *et al.*, 2003a). This implies that the presented optimal policy would more likely lead to the AM3R structure in wind turbine maintenance problems.

IV.5. Algorithm

In Section IV.3, we introduced the pure recursive technique to find the optimal policy. Now, using the structural policies developed so far, we present the new algorithm which can reduce the computational efforts substantially.

Algorithm IV.1. Construction of an optimal policy for the static CBM model

Input: C_{CM}, C_{PM}, C_{OB}, W_{CM}, W_{PM}, λ, P *and* τ.

Step 1. *Obtain* $b(e_i)$, $\forall i$ *and average cost* g *by applying policy (or value) iteration to the states on the extreme sample paths.*

Step 2. *Compute* C'_{CM}, C'_{PM} *in (4.16) and (4.17), respectively.*

Step 3. *Apply the following decision rules for a given* π.

(a) *Suppose that* $R(\pi) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$ *for* $\pi \prec_{st} \pi'(\pi)$. *If* $b_{OB}(\pi) > b_{PM}(\pi)$, $\delta^S(\pi) = PM$. *Otherwise,* $\delta^S(\pi) = OB$.

(b) *Suppose that* $R(\pi) \geq 1 - \frac{g}{C'_{CM} - C'_{PM}}$. $\delta^S(\pi) = NA$ *if* $b_{OB}(\pi) > b_{PM}(\pi)$ *or if* $R(\pi) \geq \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}$.

(c) *Suppose that* $1 - \frac{g}{C'_{CM} - C'_{PM}} \leq R(\pi) < \frac{C'_{CM} - C_{OB} - \sum b(e_i)\pi_i - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}$, *and* $b_{OB}(\pi) \leq b_{PM}$. *We apply the following recursive method, which improves the pure recursive technique.*

1. *Set* $k = 1$;
2. *If* $R(\pi^k) < 1 - \frac{g}{C'_{CM} - C'_{PM}}$, $b(\pi^k) = \min\{b_{OB}(\pi^k), b_{PM}(\pi^k)\}$. *Then apply the recursive set of equations (4.18) backward to get* $b(\pi^{k-1}), \cdots, b(\pi)$. *Otherwise,* $k = k + 1$, *and go to Step 3.*
3. *If* $||\pi^{k+1} - \pi^k|| < \epsilon$ *where* ϵ *is a small positive value, we apply (4.19) to get* $b(\pi^k)$, *and then step backwards along the path by comparing*

b_{NA} and b_{OB} to get $b(\pi^{k-1}), \cdots, b(\pi)$. Otherwise, $k = k + 1$, and go back to Step 2.

The above method results in an optimal policy that can be analytically obtained from the closed-form expressions. We need to apply the recursive method only for the states whose reliabilities are between $1 - \frac{g}{C'_{CM} - C'_{PM}}$, and $\frac{(C'_{CM} - C_{OB} - \sum b(e_i)\pi_i) - g}{C'_{CM} - C_{OB} - \sum b(e_i)\pi'_i(\pi)}$. Even for the recursive method itself, as Step 3(c).2 shows, we need not proceed until we meet the stationary state $\Pi(\pi)$. Along the sample path, once we find the state whose optimal policy is not NA (i.e. $R(\pi^k) \leq 1 - \frac{g}{C'_{CM} - C'_{PM}}$ for some π^k), we can compute $b(\pi^k)$ by comparing b_{PM} with $b_{OB}(\pi^k)$. Then we can step backwards by applying (4.18) until we get π. On the contrary, Step 3(c).3 occurs when the reliability at the stationary state is greater than $1 - \frac{g}{C'_{CM} - C'_{PM}}$. In this case, PM cannot be optimal at all of the states along the sample path originating from π. Therefore, we only need to compare $b_{NA}(\pi^k)$ with $b_{OB}(\pi^k)$ when we step backwards to π.

IV.6. Numerical examples

In this section, we present an example to illustrate the utility of the proposed dynamic maintenance policy. We examine the failures at a gearbox because gearbox problems have been identified a long while ago as one of the most serious problems in wind turbines, and the recent large-scale wind turbines with new designs still suffer badly from gearbox failures (Echavarria *et al.*, 2008, McMillan and Ault, 2008, Nilsson and Bertling, 2007, Ribrant, 2006, Ribrant and Bertling, 2007). We do want to note, however, that similar analysis can be performed for other wind turbine components as well.

IV.6.1. Problem description

We choose appropriate parameter values based on published data, and discussions with our industry partners. For the costs to repair a gearbox, we refer to Rademakers *et al.* (2003a). The total direct costs for CM, which include labor costs, crane rental, materials, and consumables, are C_{CM} =\$17,264. The PM costs are about half that of the CM costs: C_{PM} =\$8,632. We assume that the system returns to an as-good-as status after preventive maintenance. For a 2.5 MW turbine, revenue losses during one week is assumed to be τ =\$11,971. We set C_{OB} =\$1,357, according to the suggestions of our industry partners. The monetary unit of each cost factor in this example was originally in euros (Rademakers *et al.*, 2003a), but we converted from euros to the US dollar with an exchange rate of 1 euro = 1.3572 dollar.

Typical downtime after failures may take from 600 hours (25 days) up to 60 days (McMillan and Ault, 2008, Ribrant, 2006, Ribrant and Bertling, 2007). The major contribution of this lengthy down time is the long lead time when the spare parts and/or crew are not available. In this study, we assume that, upon failure, the lead time for assembling repair crew and spare parts and travel time takes six weeks. We also assume that repairs can be carried out in about one week (McMillan and Ault, 2008).

Generally, a transition matrix P can be generated from historical data by taking a long-run history about the deterioration states, and counting transitions. Due to the relatively short history of preventive maintenance practices in wind turbine industries, we do not yet have a transition matrix generated from an actual aging gearbox. So we use a P similar to the one used by Maillart (2006) with slight modifications. We will examine the sensitivity of P in the next section. We assume that the weekly-based

deterioration process follows a Markovian behavior with the following IFR matrix.

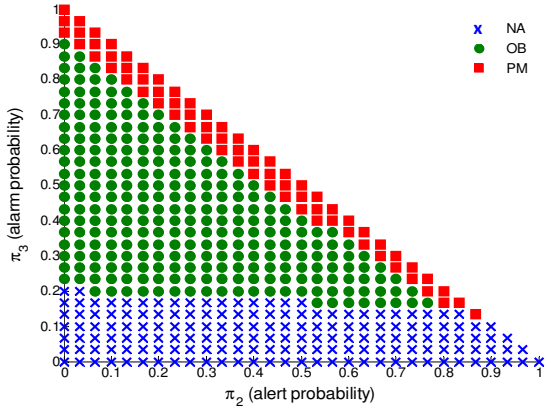
$$P = \begin{bmatrix} 0.90 & 0.05 & 0.03 & 0.02 \\ 0.00 & 0.85 & 0.10 & 0.05 \\ 0.00 & 0.00 & 0.92 & 0.08 \\ 0.00 & 0.00 & 0.00 & 1.00 \end{bmatrix} \tag{4.61}$$

Based on (4.61), we can represent the state of the gearbox as a four-dimensional row vector, $\pi = \{\pi_1, \pi_2, \pi_3, \pi_4\}$. The values π_1, π_2, and π_3 represent the probabilities of being in a *normal*, *alert*, and *alarm* state, respectively. The value π_4 represents the probability of being in a *failed* state.

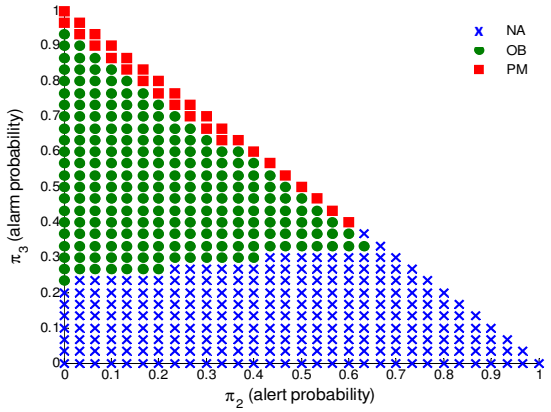
Fig. 7 illustrates the optimal policies with two different stochastic weather environments. We can see that Ω_{OB}, and Ω_{PM} are convex sets. Also, if we draw a line between any two points, the policy regions are divided into at most three regions in most cases, which is consistent with the previous discussions that the AM3R structure might dominate over the AM4R structure in real applications. Also notice that Ω_{PM} gets smaller as the chance of adverse weather conditions to prohibit PM increases. That is, with higher frequency of adverse weather conditions (i.e. with higher W_{PM}), wind farm operators should be more conservative in carrying out PM because of possible production losses caused by interrupted or delayed jobs during harsh weather.

Fig. 8 superimposes the control limits developed in Section IV.4.2 on the optimal policy for the same example in Fig. 7(a). Line 1 depicts the preference of NA to PM, or vice versa, with $R(\pi) = 1 - \frac{g}{C'_{CM} - C'_{PM}}$. Line 2 is obtained from the comparison of b_{OB} and b_{PM} with $C'_{PM} = C_{OB} + \sum_i b(e_i)\pi_i$. Finally, Line 3 defines the area where NA is preferred to OB. The optimal policy of each area is as follows:

- PM in states above Line 1, and Line 2 (by Theorem IV.1).
- OB in states above Line 1, and below Line 2 (by Corollary 1(a)).



(a)



(b)

Fig. 7. Optimal policy: (a) $W_{PM} = 0.1, W_{CM} = 0.4$ (b) $W_{PM} = 0.4, W_{CM} = 0.4$

- NA in states below Line 1, and Line 3 (by Lemma 2, and Lemma 3).
- NA in states in the triangular area surrounded by Line 1, Line 2, and Line 3 (by Corollary 1(b)).

The only states whose optimal policy are not straightforward from these control limits are shown in the region surrounded by the dashed lines in Fig. 8. The optimal policy in this region is obtained by applying the improved recursive technique discussed in Section IV.5.

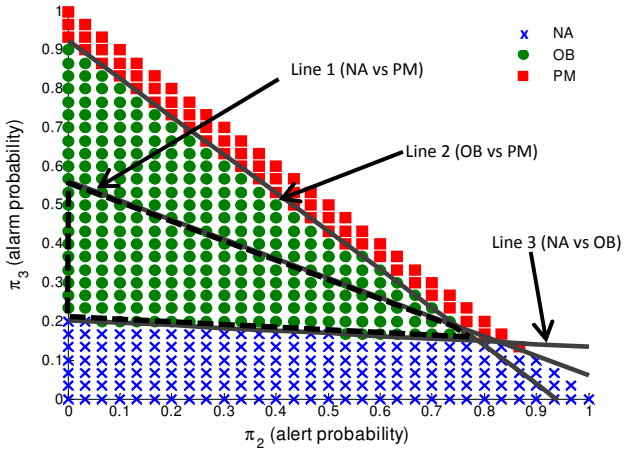


Fig. 8. Control limits superimposed on optimal policy for $W_{PM} = 0.1, W_{CM} = 0.4$

IV.6.2. Performance comparison

Suppose that we want to find the optimal policy at every grid point, as shown in Fig 7. As the dimension of states increases, computation time significantly increases when we use the pure recursive algorithm. Fig. 9 compares the performance of the suggested algorithm with the pure recursive technique. In the figure, "$M+1$" denotes the dimension of a state and "# states" denote the number of states that we evaluate to optimal policies. We use the same parameter values in the previous examples, and $W_{PM} = 0.1$ and $W_{CM} = 0.4$, but vary the transition matrix along a state size. The results indicate that the closed form of decision boundaries compounded by the improved recursive technique reduces the computation time as much as 70%, especially over the large-sized problem instances.

IV.6.3. Sensitivity analysis of transition matrix

Considering difficulties to get a transition matrix P, we analyze the sensitivity of a transition matrix by applying four additional, different matrices, P_i, $i = 1, \cdots, 4$. P_1 represents a more slowly deteriorating system in a stochastic sense than P in

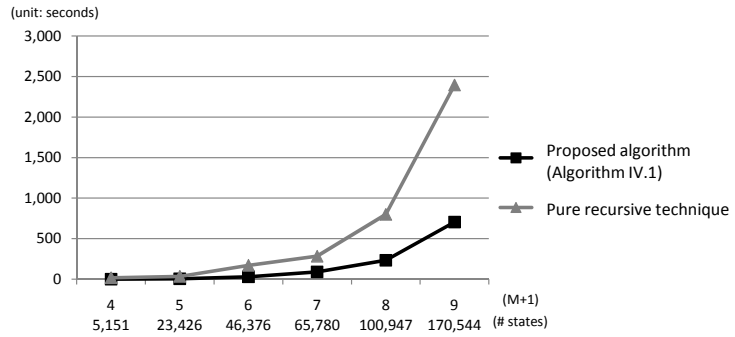


Fig. 9. Performance comparison

(4.61). That is, each row vector of P_1 is stochastically less than the corresponding row vector of P. Let us denote this relationship by $P_1 \prec_{st} P$. Similarly, $P_2 \prec_{st} P$, and also $P_1 \prec_{st} P_2$. On the other hand, P_3, and P_4 represent more rapidly deteriorating systems than P, such that $P \prec_{st} P_3 \prec_{st} P_4$.

We quantify the speeds of deterioration of a system with P_k, $k = 1, \cdots, 4$, compared to P, with the measure

$$\Delta P_k(\%) \;=\; \sum_{i=1}^{M} \sum_{j \geq i} \frac{|P(i,j) - P_k(i,j)|}{P(i,j)} \times 100, . \tag{4.62}$$

where $P(i,j)$ is the element in i^{th} row and j^{th} column of P matrix, and $P_k(i,j)$ is similarly defined. Note that the lower off-diagonal elements are not involved in (4.62) because we consider upper-triangular matrices. ΔP_k implies the relative difference of P_k, compared to P.

To measure the sensitivity of a transition matrix, we use simulation. Suppose that the actual system undergoes a deterioration process following a transition matrix P. We simulate the trajectories of system states following P from 136 different starting points. Here, 136 starting points are the points in the grid, similar to the grid points shown in Fig. 4. To speed up the simulations, we use a coarser grid such that the distance between adjacent grid points is $\frac{2}{3}$. From each starting point, the simulation

is performed over 1,000 periods. At each period, we take actions as the optimal policy suggests. Then the costs are averaged. This process is repeated 30 times. That is, we gain the average cost g by the simulations on 136 different starting points $\times$ 1,000 periods $\times$ 30 trajectories (runs).

Suppose that we do not know the transition matrix exactly, so we incorrectly use the transition matrix P_k to attain optimal policies, while the actual deterioration process follows P. We apply the similar simulation process, but we use P_k to decide the optimal policy. Then, we compute the average cost g_k. From the results of the simulations, we quantitatively measure the sensitivity of each transition matrix by

$$\Delta G_k = \frac{g_k - g}{g} \times 100. \tag{4.63}$$

Table 2 summarizes the results. The fourth column (i.e. g_k) shows that the average costs increase as the assumed transition matrix P_k deviates from the actual transition matrix P. However, the difference is not significant, as the fifth column (i.e. ΔG_k) indicates. When the values of the actual transition matrix deviates from the assumed transition matrix values by about 10% such as P_1 and P_4, the increased cost is about 2.0% on average. When the element values are different by 5-6% such as P_2 and P_3, average costs are increased by around 1%.

Although the results show that the average costs are not seriously affected by the deviation of the assumed transition matrix from the actual one, we recommend making considerable efforts to accumulate data regarding system deterioration. Rademakers *et al.* (2003a) also suggest that industry parties should share data for the improvement of O&M for wind turbines. For conventional power systems, these data for critical equipment such as circuit breakers and transformers have been accumulated, and several preventive maintenance strategies have been introduced based on historical data (Endrenyi *et al.*, 2001, 1998, Sotiropoulos *et al.*, 2007).

Table 2. Sensitivity analysis on P

P_k	ΔP_k	g	g_k	ΔG_k
P_1	10.3%	3460.0	3527.9	2.0%
P_2	6.1%	3460.0	3496.7	1.1%
P_3	5.7%	3460.0	3491.8	0.9%
P_4	10.1%	3460.0	3523.7	1.9%

Similar efforts are necessary in wind power industries.

CHAPTER V

DYNAMIC CBM MODEL: A POMDP MODEL WITH HETEROGENEOUS PARAMETERS

In this chapter, we develop the dynamic CBM model which extends the static CBM model presented in Chapter IV by incorporating more practical aspects of wind turbine operations. We formulate the problem as a finite-horizon model with heterogeneous parameters, and devise a backward dynamic programming to solve for the optimal policy numerically.

V.1. Model formulation

Weather conditions not only affect the generating capacity of wind farms but also determine its accessibility for major repairs. Harsh weather conditions could cause the repairing interruption and delay. The relatively long duration of a turbine-related repairing session in turn increases the chance that a repair is interrupted by adverse weather conditions. Moreover wind power generations are maximized in high wind speed seasons, downtime during these seasons could lead to huge productivity loss.

Taking such a seasonality factor into consideration, we examine the dynamic weather conditions that could significantly differ season by season. We allow the heterogeneity of parameters in this model. Therefore, the weather-related parameters, $W_{CM(l),n}$, $W_{PM(m),n}$ and τ_n, $\forall l, m$ depend on the choice of a period. Also, let us introduce a discount factor β, $0 \leq \beta \geq 1$. When $\beta = 1$, the model is the average cost model, whereas when $\beta < 1$, the model is discounted model. At each decision epoch, there are $M+1$ possible action alternatives: NA, $PM(1), \cdots, PM(M-1)$, and OB.

Let $J_n(\pi)$ denote tht total cost-to-go where n periods are left until the terminal

period. When NA is selected at the current state π, the total cost-to-go is as follows:

$$NA_n(\pi) = \sum_{l=1}^{L} \left(\tilde{\tau}_n(l) + \beta^{\lambda(l)+1} CM_{n-\lambda(l)-1}(e_{M+l})\right) H_l(\pi) + J_{n-1}(\pi'(\pi))R(\pi) \quad (5.1)$$

where

$$\tilde{\tau}_n(l) = \left(\sum_{t=1}^{\lambda(l)} \beta^t \tau_{n-t}\right) \cdot 1(\lambda(l) > 0) + 0 \cdot 1(\lambda(l) = 0), \quad (5.2)$$

and

$$\begin{aligned} CM_n(e_{M+l}) =& W_{CM(l),n} \left(\tau_n + \beta CM_{n-1}(e_{M+l})\right) + \\ & (1 - W_{CM(l),n})(\tau_n \cdot 1(\mu(l) = 1) + C_{CM(l)} + J_{n-\mu(l)}(e_1)) \end{aligned} \quad (5.3)$$

In (5.1), the first term $\tilde{\tau}_n(l)$ is the total revenue losses during the lead time, upon a system failure. If the system fails and the lead time is non-zero (i.e. $\lambda(l) > 0$), the wind farm would lose the revenue of $\sum_{t=1}^{\lambda(l)} \beta^t \tau_{n-t}$, as shown in the first component of (5.2). Note that these revenue losses depend on weather conditions, which indicates that if the system fails during the windy seasons and the failure requires long lead time, one should expect significant production loss. On the contrary, the second term in (5.2) implies the cases of minor failures with *zero* lead time.

$CM_n(e_{M+l})$ in (5.3) reflects the CM costs for the l^{th} failure mode. The first component is the expected costs caused by delays due to harsh weather conditions, which would occur with probability $W_{CM(l),n}$. The second component indicates the repair costs under good weather conditions. Note that $\tau_n \cdot 1(\mu(l) = 1)$ in (5.3) specifies the revenue losses during a major repair that takes one full period. After the repair, the system condition is restored to the best condition e_1.

Next, let us consider the actions of PM. $PM(m)$ action, $m = 1, \cdots, M-1$ can be categorized into minor repairs and major repairs in a broad sense. We assume

that the repair time for minor repairs is negligible. Therefore, minor repairs can be carried out almost instantaneously as long as the weather conditions are good (This is a more realistic setting than the one used in the static CBM model. Recall that in static CBM model, we assume that $PM(m)$ takes one full period regardless of the repair level). But, if the weather conditions are not good during the whole period, NA is taken. On the other hand, major repairs take one full period, and if the weather conditions become harsh during the repair, the job has to be halted and will be resumed in the next period. The following formulation in (5.4) is the total cost-to-go for $PM(m)$ for $m = 1, \cdots, M-1$:

$$PM_n(m) = \begin{cases} W_{PM(m),n} NA_n(\pi) + (1 - W_{PM(m),n})(C_{PM(m)} + J_n(e_m)) \\ \qquad\qquad \text{for minor repairs} \\ W_{PM(m),n}(\tau_n + PM_{n-1}(m)) + (1 - W_{PM(m),n})(\tau_n + C_{PM(m)} + J_{n-1}(e_m)) \\ \qquad\qquad \text{for major repairs} \end{cases} \tag{5.4}$$

Finally, the observation costs can be formulated in the same way as we did in the static CBM model. That is,

$$OB_n(\pi) = C_{OB} + \sum_{i=1}^{M} Post_n(e_i)\pi_i \tag{5.5}$$

where

$$Post_n(e_i) = min\{NA_n(e_i), PM_n(1), \cdots, PM_n(M-1)\} \tag{5.6}$$

Now, the optimal value function can be written as follows:

$$J_n(\pi) = min\{NA_n(\pi), PM_n(1), \cdots, PM_n(M-1), OB_n(\pi)\} \tag{5.7}$$

Solving the optimization in (5.7) gives the the optimal decision rule $\delta_n^D(\pi)$ at the current state π where the superscript D implies a dynamic policy. $\delta_n^D(\pi)$ will take one of the possible maintenance actions, $NA, PM(1), \cdots, PM(M-1), OB$, specifying the best action selection when the system occupies the state π at a specified decision epoch n. The optimal policy at the state π, denoted by $\Delta^D(\pi)$, is the vector of the optimal decision rules to be used through decision epochs, i.e. $\Delta^D(\pi) = (\delta_1^D(\pi), \cdots, \delta_n^D(\pi))$.

V.2. Proposed solution: backward dynamic programming

In order to attain the optimal policy and optimal value, we use a backward dynamic programming (Puterman, 1994). Let us consider a sample path starting from a state π, $\{\pi, \pi^2, \cdots, \Pi(\pi)\}$. Observing from (5.1) to (5.6), one can find that the total cost-to-go associated with each possible action as well as the optimal value $J_n(\pi^k)$ at π^k and period n are only dependent on the values at the next state π^{k+1} and the extreme states e_i, $i = 1, \cdots, M+L$. Utilizing this understanding, we can step backwards along the path, recursively solving for the corresponding optimal action. The following algorithm summarizes the solution procedure that finds the optimal polices along a sample path. We also provide an overview of the algorithm in Fig. 10. Let $\lambda = max\{\lambda(l); l = 1, \cdots, L\}$.

Algorithm V.1. Backward dynamic programming algorithm

Input: π, $\beta, P, C_{CM(l)}, C_{PM(m)}, C_{OB}, \lambda(l), \mu(l), \tau_n, W_{CM(l),n}$, $W_{PM(m),n}$, $\forall l, m, n$

Step 1. *Construct a sample path* $\Psi(\pi) = \{\pi, \pi^2, \cdots, \Pi(\pi)\}$, *emanating from* π. *Similarly, generate the extreme sample paths,* $\Psi_{e_1}, \cdots, \Psi_{e_M}$, *originating from the extreme states* e_m, $m = 1, \cdots, M$.

Step 2. *Set the terminal values* $J_{\lambda+1}(\pi)$ *according to the business situations (Alter-*

natively, the terminal values can be set arbitrarily for large n).

Step 3. *Repeat for* $t = \lambda + 2, \lambda + 3, \cdots, n$*:*

3.1. *Set the time-varying parameter values such as* $W_{CM(l),t}$*,* $W_{PM(m),t}$ *and* τ_t*,* $l = 1, \cdots, L$*,* $m = 1, \cdots, M-1$*.*

3.2. *Find the optimal decision rule and optimal value at each extreme point* e_i*,* $i = 1, \cdots, M$*. i.e. compute* $NA_t(e_i), PM_t(1), \cdots, PM_t(M-1)$*, for* $i = 1, \cdots, M$*. Then, find* $J_t(e_i) = min\{NA_t(e_i), PM_t(1), \cdots, PM_t(M-1)\}$ *and the corresponding* $\delta_t^D(e_i)$*,* $i = 1, \cdots, M$*.*

3.3. *Compute the total cost-to-go* $CM_t(e_{M+l})$ *for each* CM *with the* l^{th} *failure mode,* $l = 1, \cdots, L$*.*

3.4. *For* $\forall \pi^k \in \Psi_\pi$*, compute the total cost-to-go associated with each action,* $NA_t(\pi^k)$*,* $PM_t(1)$*,* $\cdots$*,* $PM_t(M-1)$*,* $OB_t(\pi^k)$

3.5. *Get the optimal value function* $J_t(\pi^k)$*,* $\forall \pi^k \in \Psi_\pi$ *and the corresponding optimal decision rule.*

3.6. *Set* $t = t + 1$*, and go back to Step 3(a).*

At Step 2 of the algorithm, one option to assign the terminal value is to use the salvage value of the component. Alternatively, we can set the terminal values *arbitrarily* since the terminal value would not affect the optimal decision rules at the initial periods when n is large enough. Without loss of generality, $J_{\lambda+1}(\pi) = 0$ is used, $\forall \pi$.

We evaluate the optimal values at the extreme states at Step 3.2, before evaluating the optimal values at other non-extreme states at Step 3.4 and 3.5. Note that for calculating the optimal values for the extreme states, OB is not considered as

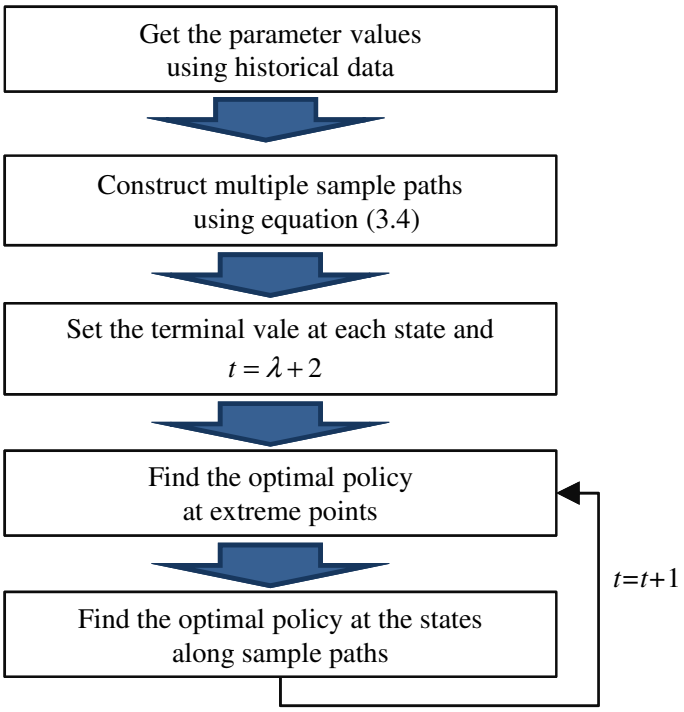


Fig. 10. Overview of the proposed backward dynamic programming algorithm

one of the potential optimal actions for selection because we know that OB cannot be optimal at the extreme points (see (5.5) and (5.6)). But, in order to compute $OB_n(\pi^k)$ at the non-extreme states, we need to know $V(e_i)$'s, $i = 1, \cdots, M$, which explains why Step 3.2 comes first before Step 3.4 and 3.5.

Since the weather related parameters (i.e. $W_{CM(l),n}$, $W_{PM(m),n}$, τ_n) are season-dependent, the above procedure generates a *non-stationary* optimal policy, making the policy dynamically adjusted to seasonal effects.

V.3. Case study

As explained in the previous chapter, most critical failures in wind turbines are associated with the gearbox because of high capital cost, long lead time for repairs, difficulty in replacing a gearbox, and lengthy downtime compounded by adverse

weather conditions (Echavarria *et al.*, 2008, McMillan and Ault, 2008, Nilsson and Bertling, 2007, Ribrant, 2006, Ribrant and Bertling, 2007). Therefore, we again choose a gearbox among several components of a wind turbine to illustrate the presented methodology in this chapter.

V.3.1. Problem description

We assume that the wind farm operators make maintenance decisions on a weekly basis. Appropriate parameter values are selected based on the published data or discussions with our industry partners.

Ribrant (2006) and Ribrant and Bertling (2007) examine the failure frequencies of different failure modes and the corresponding downtime in gearboxes of wind turbines with a rated power of $490kW$ or more. Table 3 summarizes the statistics related to the gearbox failures. The failures of bearings and gearwheels often demand a total change of the gearbox, resulting in a long downtime. The unspecified failure types in the fourth row of Table 3 sometimes correspond to other serious failures which require a replacement of the whole gearbox (Ribrant, 2006). The other two failure modes require minor repairs in general. The first three columns are obtained from Ribrant (2006), Ribrant and Bertling (2007). Based on these numbers, we set the lead time and repair time for each failure type, as shown in the fourth and fifth columns.

As explained in the previous chapters, a transition matrix P can be obtained from operational data by taking a long-run history about the degradation states and counting transitions. For critical equipment such as circuit breakers and transformers in the conventional power systems, aging-related data have been accumulated for a long time, and several repair strategies have been presented using a Markov process (Billinton and Li, 2004, Hoskins *et al.*, 1999, Jirutitijaroen and Singh, 2004, Qian

Table 3. Failure types of a gearbox

Sub-component[a]	failure frequency[a]	Average (min-max) downtime (hours)[a]	lead time (weeks)[b]	repair time (weeks)[b]	Corresponding repair action
Oil systems	11.9%	26 (1-63)	0	0	$CM(1)$
Sealing	7.3%	52 (2-218)	0	0	$CM(2)$
Not specified	40.4%	230 (9-1,248)	1	1	$CM(3)$
Gearwheels	2.8%	272 (57-383)	1	1	$CM(4)$
Bearings	37.6%	562 (15-2,067)	2	1	$CM(5)$

[a] The data are obtained from Ribrant (2006) and Ribrant and Bertling (2007)
[b] The lead time and repair time associated with each failure mode are set on a weekly basis for the modeling purpose

et al., 2007, Welte, 2009, Yang *et al.*, 2008). For the wind industry, there is a lack of data in the public domain for calculating the precise transition matrix for wind turbine components. For the time being, the common remedy researchers adopt is to use the limited amount of data, combined with expert judgments or simulations, to estimate the transition probabilities, for instance, the approach used in the study of McMillan and Ault (2008).

In this study we follow a similar approach to handle the transition probabilities as in the above-mentioned literature. We analytically derive the *first passage time* (Norris, 1998) to the failure as a function of the elements of a transition matrix, which is a MTTF in the reliability study (Jirutitijaroen and Singh, 2004). The inverse of MTTF gives the average failure frequency. Then, we apply a similar transition matrix used in Maillart (2006) and modify the matrix to be consistent with the overall failure

frequency of a gearbox and the frequency of each failure mode shown in Table 3. According to Ribrant (2006), the failure frequency of a gearbox ranges from 0.05-2.29 times per year, depending on the turbine manufacturers and models. Since most wind farm operators currently perform SchMs regularly, we believe that this failure frequency is the result under the SchM practice. Based on these understandings, we construct the transition matrix P with the following submatrices.

$$P_A = \begin{bmatrix} 0.93 & 0.04 & 0.029 \\ 0.00 & 0.95 & 0.03 \\ 0.00 & 0.00 & 0.96 \end{bmatrix},$$

$$P_B = \begin{bmatrix} 0.001 & 0 & 0 & 0 & 0 \\ 0.002 & 0.001 & 0.008 & 0.001 & 0.008 \\ 0.004 & 0.003 & 0.016 & 0.002 & 0.015 \end{bmatrix} \tag{5.8}$$

Since we consider one week as a transition period, P represents a weekly-based deterioration process. The state can be represented as an eight dimensional row vector, $\pi = \{\pi_1, \pi_2, \pi_3, \pi_4, \cdots, \pi_8\}$. π_1, π_2 and π_3 are the probabilities of being in a *normal*, *alert* and *alarm* condition, respectively, and π_4 to π_8 represent the five different failure modes, as shown in Table 3.

Remark: Using Monte Carlo simulations, we validate that the failure frequencies with P_A and P_B in (5.8) are consistent with the industry statistics under a SchM. The parameter values presented in (5.8), however, may not be a definitive set of values; rather they could be a starting point for deriving condition-based maintenance policy and evaluating the benefits of the proposed model framework. As McMillan and Ault (2008) point out, future work is needed to better quantify the parameter values in P. Rademakers *et al.* (2003a) also suggest that industry parties should collaborate with one another to collect and share data for the improvement of wind farm O&M. It also

should be emphasized that a much refined definition of system conditions allowing more levels of possible PM actions may be necessary in real situations while we only consider these three levels of system conditions in this case study. Doing so will need to use an information state π of a higher dimension, but the proposed methodology can be similarly applied.

There are five types of corrective maintenances, $CM(1), \cdots, CM(5)$ and two types of preventive maintenances, $PM(1)$ and $PM(2)$. To get the maintenance costs, we refer to Andrawus *et al.* (2007). Rademakers *et al.* (2003a) also discuss different cost factors in their study. According to Andrawus *et al.* (2007), a major CM, as a result of unanticipated failures, costs \$122,787, and a major preventive repair costs \$12,820. Therefore, we set $C_{CM(4)}$ and $C_{CM(5)}$ to be \$122,787 and $C_{PM(1)}$ to be \$12,820 because $CM(4)$, $CM(5)$ and $PM(1)$ correspond to major repairs. There are no cost figures for $C_{PM(2)}$ in the literature, and not for $C_{CM(1)}$ through $C_{CM(3)}$, either. So based on the suggestions of our industry partners, we set $C_{CM(3)}$ to be the half of the major CM cost, and $C_{CM(1)}$ and $C_{CM(2)}$ to be one tenth of the major CM cost, respectively. Similarly, $PM(2)$ corresponds to the minor repair and its cost $C_{PM(2)}$ is assumed to be one third of the major PM cost. The OB cost of a gearbox is set to be \$313.36 (Andrawus *et al.*, 2007). The monetary unit of each cost factor in this example was originally in pounds(£) (Andrawus *et al.*, 2007), but we converted from pounds to the US dollar with an exchange rate of 1 pound = 1.5668 dollar. These costs are summarized in the second column of Table 4.

Furthermore, these maintenance activities are constrained by the weather conditions. The weather conditions would depend on the locations, terrains of the wind farm site. We set the probabilities that the harsh weather conditions would occur each season in the third column of Table 4.

Revenue losses per period depend on the weather conditions. We set the average revenue losses to be \$6,946 (Andrawus *et al.*, 2007). Then, we adjust the revenue losses across the four seasons from spring to winter to be 80%, 120%, 80% and 130% of the average revenue losses, respectively. Table 5 summarizes the potential revenue losses per week for each season.

Table 4. Maintenance costs and harsh weather probabilities for each maintenance action

Repair types[a]		Repair costs[b]	Weather parameters			
			Spring	Summer	Fall	Winter
Minor CM	$CM(1)$	12,279	0.05	0.2	0.05	0.2
	$CM(2)$	12,279	0.05	0.2	0.05	0.2
Major CM	$CM(3)$	61,394	0.1	0.3	0.1	0.4
	$CM(4)$	122,787	0.1	0.4	0.1	0.6
	$CM(5)$	122,787	0.1	0.4	0.1	0.6
Major PM	$PM(1)$	12,820	0.1	0.3	0.1	0.4
Minor PM	$PM(2)$	4,273	0.05	0.2	0.05	0.2

[a] Major repairs take one week, whereas the duration for minor repairs is negligible.
[b] The monetary unit is the US dollar.

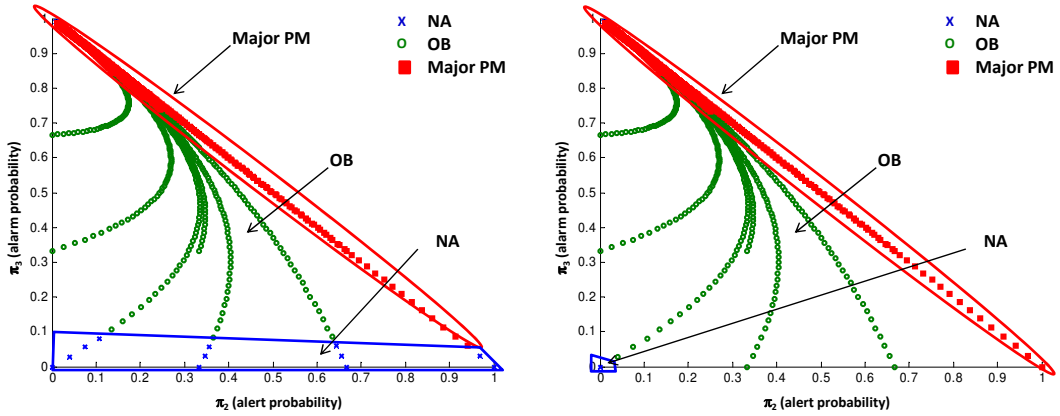
V.3.2. Results from optimal policy

With these parameter values, we compute the optimal policy during a 20-year decision horizon. Since we consider the decision-makings on a weekly basis, we set the discount

Table 5. Revenue losses

Spring	Summer	Fall	Winter
5,556	8,335	5,556	9,029

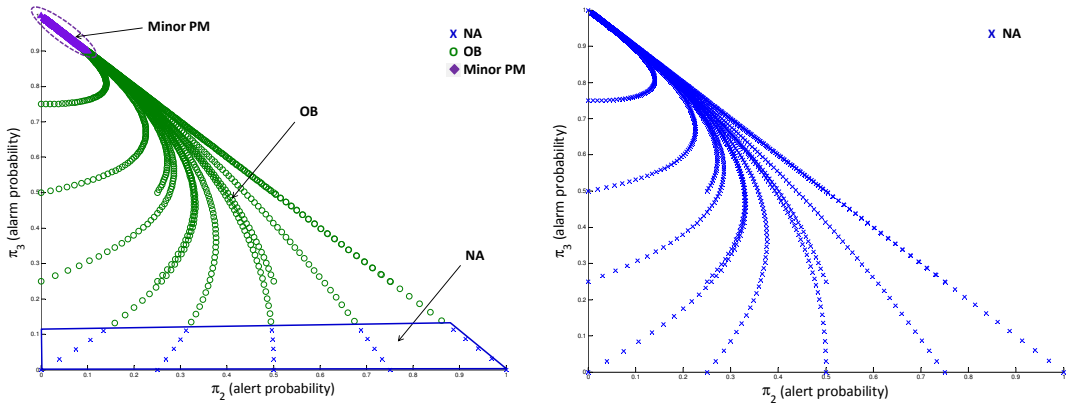
factor β as 0.99, which is close to *one*. Fig. 11 through Fig. 13 show, respectively, the optimal actions for spring, summer and fall seasons in the first year of operations along a number of series of the sample paths. The optimal actions for winter season are almost similar to the results for summer season.



(a) In the beginning of spring (b) At the end of spring

Fig. 11. Optimal decision rule during spring season

The resulting policy in the figures can be understood as follows: the viable operating region (the triangle in Fig. 5 in Section III.3) can be partitioned into subregions corresponding to different actions (i.e. NA, OB, major PM, and minor PM). It is noticeable that each curve in Figs. 11-13 (except Fig. 12(b)) has a couple of different colors (and shapes) in order to specify different optimal policies along

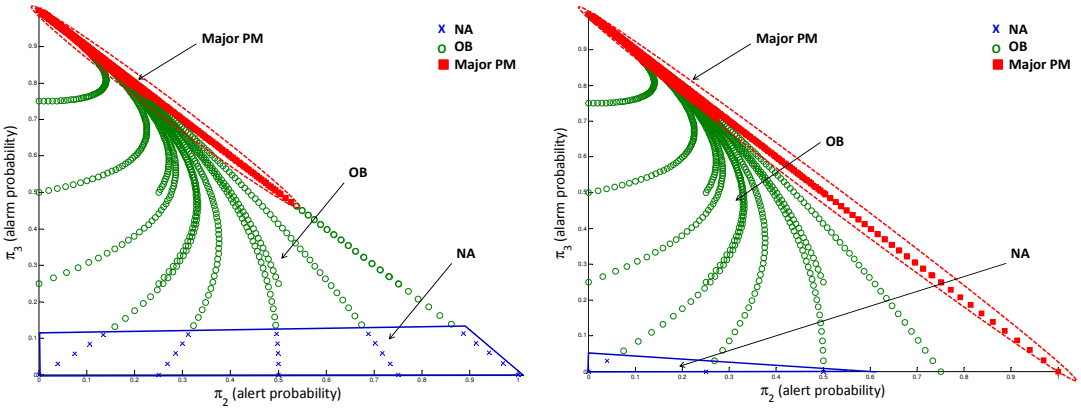


(a) In the beginning of summer (b) At the end of summer

Fig. 12. Optimal decision rule during summer season

a sample path. For example, in Fig. 11(a), one curve originating from the origin consists of markers of three different colors and shapes: first the blue X's, then the green O's, and then, the red squares. It implies that wind farm operators should take no action for the first segment in this curve when a state belongs to the NA region, and it becomes optimal to take OB in the second segment until the curve reaches the region where PM is optimal.

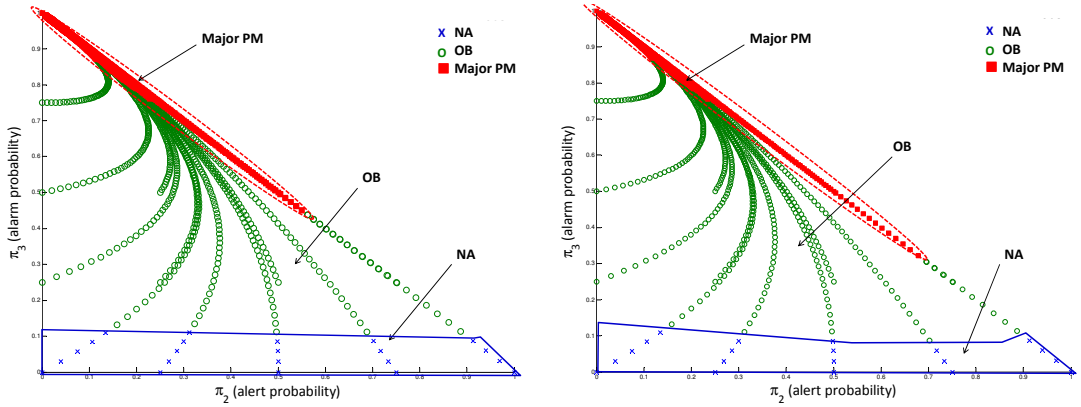
By obtaining the optimal policies along multiple sample paths, we can easily identify each region where a specific action is optimal. Then, at the beginning of each decision period, wind farm operators just need to estimate the system states (π_2 and π_3, the horizontal and vertical axes in the figure) and to check which subregion the state estimate falls in, and then take actions according to the corresponding type of that subregion. For example, in the beginning of spring (See Fig. 11(a)), if a state falls in a major PM area in the upper-left corner surrounded by the red dashed boundary, wind farm operators should take major preventive repairs. Neither repair nor observation is required in the case that a state falls in the NA area in the lower part surrounded by solid lines. Understandably, if a state falls in the area in between,



(a) In the beginning of fall (b) At the end of fall

Fig. 13. Optimal decision rule during fall season

wind farm operators should take OB.



(a) In the middle of spring (b) In the middle of fall

Fig. 14. Optimal decision rule in the middle of spring and fall

It is interesting to see that the optimal policy is non-stationary. That is, the optimal action is not the same throughout the decision periods. It is worth noting a few features of the optimal decision rules.

Observation 1. In the beginning of mild weather seasons such as spring and fall, we take the major PM's when the system is estimated to be ill-conditioned. Toward the end of mild seasons, the optimal decision is to take the major PM's even for

the moderately deteriorated condition like e_2 in order to minimize the risk of failures during the upcoming harsh weather seasons.

Observation 2. The optimal decisions of spring and fall seasons are slightly different. The NA area at the end of fall season in Fig. 13(b) is smaller than the one in Fig. 11(b). Fig. 14 compares the two optimal policies in the middle of spring and fall seasons. The area where the major PM is optimal in Fig. 14(b) is larger than the area in Fig. 14(a). All these are because of the more restricted maintainability of the wind turbines during the (almost entire) winter season.

Observation 3. In the beginning of harsh weather seasons such as the summer storm season and winter season, it is recommended to take the minor PM's for the seriously ill-conditioned system to avoid failures during the remaining harsh weather periods, occurring of which may cause tremendous repair costs. However, at the end of harsh weather seasons, NA is dominated in all the states because it would be better to wait for the next mild periods rather than performing risky repair activities right away.

Observation 4. OB is taken when the system conditions are not clear. However, OB is taken more often in the beginning or middle of harsh weather seasons to decide the most suitable maintenance tasks than in the mild seasons. Understandably, doing so will help reap more economical benefits.

V.3.3. Comparison of different maintenance strategies

To quantify the benefits of the proposed dynamic CBM strategy, we compare the optimal policy with two other maintenance strategies. The first strategy is the SchM with fixed, regular repair schedules, reflecting the current industry practices. The other strategy is a similar CBM strategy, but without considering the seasonal weather effects.

To compare each strategy, we conduct Monte Carlo simulations using the same parameter values explained in Section V.3.1. We simulate the system states using the transition matrix with P_A and P_B in (5.8). We also simulate the weather scenarios with the given probabilities in Table 4. For each strategy, 30 trajectories (runs) of simulations are performed over 1,040 periods (= 52 weeks × 20 years). Then, we obtain the average failure frequency and O&M costs per year. Table 6 and Fig. 15 summarize the simulation results of each maintenance strategy, and we will explain the implications of the results.

Table 6. Average of simulation results for different maintenance strategies (standard deviation in parenthesis)

	SchM	Static CBM	Dynamic CBM
# failures per year	1.29 (0.31)	0.74 (0.30)	0.55 (0.16)
O&M costs per year[a]	107,044 (21,167)	102,923 (14,133)	90,859 (13,945)

[a] The monetary unit is the US dollar.

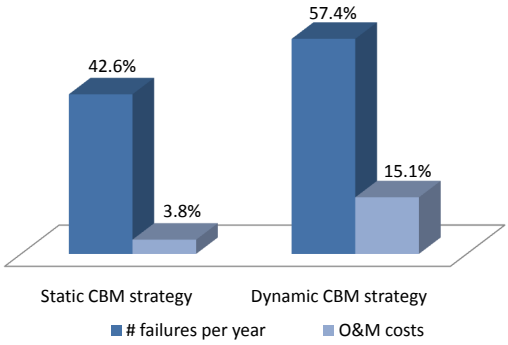


Fig. 15. Reduction (%) of failure frequency and maintenance costs of the two CBM strategies compared with the current industry practices

Results from current industry practices. Current industry practices are mainly based on the SchMs, which conducts PM's on a regular basis in low wind

speed seasons (Nilsson and Bertling, 2007, Walford, 2006). The frequency of the SchM usually depends on the manufacturer's recommended maintenance program (Pacot *et al.*, 2003). However, according to Nilsson and Bertling (2007), wind farm operators usually carry out minor maintenances twice a year and major maintenances once every 2 to 4 years, respectively. Following the industry practices, we set the scheduled minor maintenances to be carried out in spring and fall, and major preventive maintenances to be performed once every three years in spring. The simulation results indicate that generators would fail 1.29 times per year on average, resulting in $107,044 costs per year under this SchM strategy.

Results from the static CBM strategy. Suppose that in order to produce a condition-based maintenance policy, wind farm operators consider a gearbox's degradation status but ignore the weather constraints. That is, the maintenance policy is obtained with the assumption that maintenance tasks can be performed anytime although repair tasks are constrained by seasonal weather effects.

To implement this strategy, we set $W_{CM(l),n}$'s and $W_{PM(m),n}$'s to be *zero*, and use a constant $\tau_n, \forall l, m, n$ in the Algorithm V.1. Then, the resulting decision rules under the assumption of static weather conditions are applied at each period in the simulation. This strategy is similar to the static CBM model introduced in Chapter IV, in the sense that homogeneous weather-related parameters are used, and thus, the resulting strategies are static over the decision horizon (but the difference is that the static CBM model use non-zero constants for $W_{CM(l),n}$'s and $W_{PM(m),n}$'s).

Fig. 16 illustrates the optimal decision rules under this static CBM strategy. With this strategy, wind farm operators take the action based on the degradation state of a gearbox, but the same maintenance action will be applied to the same state over the different seasons. The third column of Table 6 summarizes the results from this

strategy. Since this strategy considers the deterioration status, one can make timely decisions regarding when to take maintenance actions to avoid failures. As a result, the failure frequency is reduced by 42.6% (=(1.29-0.74)/1.29), compared with the result of the SchM. However, the reduction of O&M costs comes at an unimpressive 3.8% (=(107,044-102,923)/107,044) since this strategy does not consider the weather impacts (See the graph in the left side of Fig. 15).

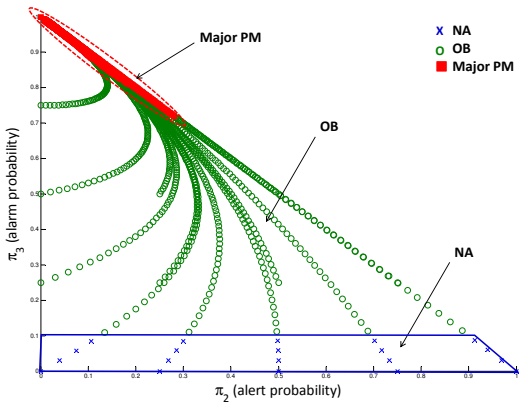


Fig. 16. Optimal decision rule under stationary weather conditions

Results from the dynamic CBM strategy. In this strategy, the optimal maintenance action suggested by this chapter, which considers the costs, degradation status and weather conditions, is taken at each decision period. The final column of Table 6 summarizes the results from the optimal policy. The reductions in both failure frequency and O&M costs are remarkable, compared with the SchM. The failure frequency and O&M costs are decreased by 57.4% (=(1.29-0.55)/1.29) and 15.1% (=(107,044-90,859)/107,044), respectively, demonstrating that substantial benefits can be anticipated by adopting the proposed dynamic CBM strategy in the practices of wind power industry (See the graph in the right side of Fig. 15).

CHAPTER VI

SIMULATION OF WIND FARM OPERATIONS USING DEVS

In this chapter, we describe the simulation model for a wind farm for predicting wind turbines' states and assessing maintenance actions. We use the DEVS formalism (Zeigler *et al.*, 2000) to derive the models that can be tailored to any real wind farm. In our simulation model, we consider the maintenance of a *gearbox* among several components of wind turbines to illustrate the simulation model. We implement two different O&M strategies: SM and CBM. Here, the CBM strategy is different from the strategies discussed in Chapters IV and V. Rather, we implement a simpler CBM strategy which has been recently adopted in practice (Pacot *et al.*, 2003).

VI.1. Model abstraction

VI.1.1. Power generation model

Power generated from wind turbines mainly depends on wind speed and can be calculated using a power curve as shown in Fig. 17 (Karki and Patel, 2009). Wind turbines are designed to start generating power at the cut-in wind speed WS_{ci}. The power output increases nonlinearly as the wind speed increases from the cut-in wind speed to the rated wind speed WS_r. However, at higher wind speeds than the cut-out wind speed WS_{co} turbines are shut down to avoid damage to the structure due to excessive mechanical loads. These parameter values depend on the type of wind turbine and are usually specified by wind turbine manufacturers.

According to (Karki and Patel, 2009), the mathematical relationship between

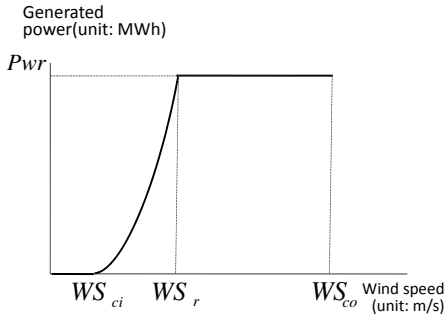


Fig. 17. Power curve (Karki and Patel, 2009)

the wind speed WS and the generated power Pwr can be given as follows:

$$Pwr = \begin{cases} 0, & \text{if } 0 \leq WS < WS_{ci} \\ P_r(a + b \cdot WS + c \cdot WS^2) & \text{if } WS_{ci} \leq WS < WS_r \\ P_r, & \text{if } WS_r \leq WS < WS_{co} \\ 0, & \text{if } WS_{co} \leq WS, \end{cases} \tag{6.1}$$

where P_r is the rated power output of the wind turbine. The parameters a, b, c in equation (6.1) are obtained by the following equations:

$$a = \frac{1}{(WS_{ci} - WS_r)^2} \left[WS_{ci}(WS_{ci} + WS_r) - 4WS_{ci}WS_r \left(\frac{WS_{ci} + WS_r}{2WS_r} \right)^3 \right] \tag{6.2}$$

$$b = \frac{1}{(WS_{ci} - WS_r)^2} \left[4(WS_{ci} + WS_r) \left(\frac{WS_{ci} + WS_r}{2WS_r} \right)^3 - (3WS_{ci} + WS_r) \right] \tag{6.3}$$

$$c = \frac{1}{(WS_{ci} - WS_r)^2} \left[2 - 4 \left(\frac{WS_{ci} + WS_r}{2WS_r} \right)^3 \right] \tag{6.4}$$

VI.1.2. Wind speed model

When anemometers are installed inside turbines, wind speed models at turbine locations can be built using temporal models. However, when wind speeds are measured

at stations near wind turbine sites, the spatial correlations between wind speeds at the stations and those at the wind turbine sites have to be considered. The latter case happens when evaluating a new wind farm site where the wind speeds are not available but the wind speeds at the stations near the target wind farm site are available. This is also true when wind turbines do not have anemometers, which can be encountered often in old, small-sized wind turbines. A spatio-temporal model that considers both temporal and spatial variations of wind speeds is required in this case.

We develop a spatio-temporal model in a hierarchical manner to generate synthetic sequences of wind speeds at turbine locations. This hierarchical construction makes the model flexible so that when the wind speeds are available at the turbine site, we only need the temporal model. The generated sequences should represent possible realizations of wind speeds at the turbine sites. Therefore, each of the components included in a spatio-temporal model has a physical interpretation and include diurnal cycle, yearly seasonality, and the spatial variations among different locations.

First, we build a time series model at each station which measures wind speeds on a regular basis. In the model, diurnal cycle and yearly seasonality are modeled using Fourier series (De Luna and Genton, 2005, Gneiting *et al.*, 2007). Let us denote the wind speed at time t at station x_i, $i = 1, 2, \cdots, N_s$, by $S_t(x_i)$. Then, $S_t(x_i)$ at each station, $i = 1, \cdots, N_s$ can be modeled as follows:

$$\begin{aligned} S_t(x_i) &= \alpha_0^i + \sum_{d=1}^{n_d} \left[\beta_{1d}^i \sin(d\frac{2\pi t}{\omega_d}) + \beta_{2d}^i \cos(d\frac{2\pi t}{\omega_d}) \right] + \\ &\quad \sum_{s=1}^{n_s} \left[\gamma_{1s}^i \sin(s\frac{2\pi t}{\omega_s}) + \gamma_{2s}^i \cos(s\frac{2\pi t}{\omega_s}) \right] + \epsilon_t^i. \end{aligned} \tag{6.5}$$

In equation (6.5), the first term α_0^i is the average wind speed and the second term $\beta_{1d}^i \sin(d\frac{2\pi t}{\omega_d}) + \beta_{2d}^i \cos(d\frac{2\pi t}{\omega_d})$ is the d^{th} harmonic to represent the *diurnal cycle*, where ω_d is the number of wind speed measurements during a day (Soares and Medeiros, 2008).

Similarly, s in the third term $\gamma_{1s}^i \sin(s\frac{2\pi t}{\omega_s}) + \gamma_{2s}^i \cos(s\frac{2\pi t}{\omega_s})$ is the s^{th} yearly harmonic to represent yearly seasonality, where ω_s is the number of measurements during a year. In many environmental studies, n_d and n_s are typically set to 1 or 2 (Gneiting *et al.*, 2007, Im *et al.*, 2008, Magnano and Boland, 2007, Soares and Medeiros, 2008). The parameter ϵ_t^i in (6.5) is a stochastic component of the time series model, which can be formulated as the following ARMA model:

$$\epsilon_t^i - \sum_{p=1}^{D_p} \phi_p^i \epsilon_{t-p}^i = a_t^i - \sum_{q=1}^{D_q - 1} \theta_q^i a_{t-q}^i, \tag{6.6}$$

where a_t^i is an i.i.d random variable with distribution $N(0, \sigma_i^2)$. The appropriate degree (D_p, D_q) of the ARMA model can be selected by using the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC) (Bowerman *et al.*, 2005). The related parameters in the above ARMA model, ϕ_p^i, θ_q^i, and σ_i^2, $\forall p, q, i$ can be estimated by going through a generalized linear model analysis.

The wind speed at a turbine site x at time t, denoted by $W_t(x)$, can be generated considering spatial correlations between stations and turbine sites. We use a spatial modeling approach, called *Kriging* (Haining, 2000). Suppose that wind speeds at the stations, $S_t(x_1), \cdots, S_t(x_{N_s})$ are simulated (or observed) at spatial locations $x_1, \cdots, x_{N_s}$ at time t. Then, the wind speed at a wind turbine site x at time t is given by,

$$W_t(x) = \sum_{j=1}^{J} \varphi_j f_j(x) + r^\top R^{-1}(S_t - \varphi \cdot e), \tag{6.7}$$

where $f_j(x)$, $j = 1, \cdots, J$, are known regression functions taking spatial locations x as input. In this study, we use simple Kriging with $\sum_{j=1}^{J} \varphi_j f_j(x) = \varphi$, which is constant. $S_t = [S_t(x_1), \cdots, S_t(x_{N_s})]^\top$ is the $N_s \times 1$ matrix, each element of which denotes a wind speed at each station at time t, and $e = [1, 1, \cdots, 1]^\top$ is $N_s \times 1$ vector.

R is the correlation matrix between wind speeds at stations, whose $(k,l)^{\text{th}}$ element is as follows:

$$R(x_k, x_l) = \exp\{-\sum_{u=1}^{U} |(x_{ku} - x_{lu})/\nu_u|^{\psi}\}, \tag{6.8}$$

Here, U is a dimension of x_i, $i = 1, \cdots, N_s$, and $R(x_k, x_l)$ denotes the correlation function between two locations, $x_k = [x_{k1}, \cdots, x_{kU}]^{\top}$ and $x_l = [x_{l1}, \cdots, x_{lU}]^{\top}$, $k, l = \{1, \cdots, N_s\}$. ψ is the shape parameter to represent the correlation function shape, whereas ν_u is the scale parameter to denote the effect of distance in each dimension u, $u = 1, \cdots, U$. Finally, $r = [R(x, x_1), \cdots, R(x, x_{N_S})]^{\top}$ in (6.7) is $N_S \times 1$ dimensional vector, each of element reflects the correlation between the wind turbine site and the station.

Note that $W_t(x)$ in (6.7) is the estimated wind speed at the anemometer height. If the anemometer height does not coincide with the wind turbine hub height, the wind speed has to be scaled based on the turbine hub height. There are different formulas for the wind speed adjustment (Negra *et al.*, 2007, Zhou *et al.*, 2006), among which the following *wind power law* has been recognized as a useful tool to transfer the anemometer data to the desired hub center in many studies (Gipe, 2000, Zhou *et al.*, 2006).

$$W_t^H(x) = W_t(x) \left(\frac{z}{z_0}\right)^{\alpha}, \tag{6.9}$$

where $W_t^H(x)$ is the wind speed at anemometer height z at turbine location x and $W_t(x)$ is wind speed at hub height z_0 from equation (6.7). The parameter α is a wind speed power law coefficient, whose value mainly depends on the local geographical terrain.

VI.1.3. Wind turbine components with degradation model

Although there are different designs of wind turbines, they share basic common features and the names of the components are general (Pacot *et al.*, 2003). Fig. 18 shows the different components of a wind turbine (Pacot *et al.*, 2003). A wind turbine consists of a tower, two or three-bladed rotors, and a nacelle which houses several critical components such as the drive train, gearbox, generator and the electrical system.

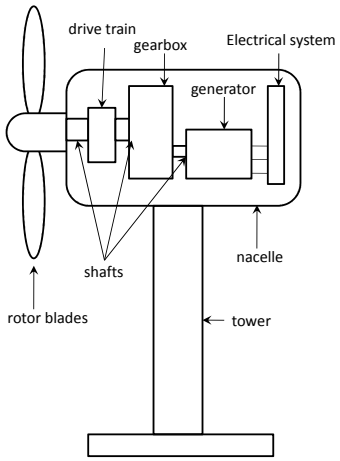


Fig. 18. Wind turbine components (Pacot *et al.*, 2003)

There are several mathematical models to represent component degradation. In this study, we use a Markov model for the consistency with the optimization models in Chapters IV and V.

VI.1.4. Sensor models

In a regular Markov model, the state is directly visible to the observer. However, we assume the state is not directly visible, but output (here, sensor output) dependent on the state is available. Suppose that one can categorize the sensor results into a finite number of outputs, $k = 1, 2, \cdots, K$. Then, the probability of getting sensor

output k when the current state is j is $SE_j(k)$, where $\sum_{k=1}^{K} SE_j(k) = 1$ for each j. As such, the sequence of sensor outputs characterizes the system degradation in a probabilistic sense.

VI.1.5. State evaluation model.

Given sensor information, we need a model for performing system state evaluation. Let us denote the current system state by Z_t at time t. Based on sensor data streams, we estimate the system state Z_t using the *Viterbi* algorithm (Qian *et al.*, 2007). Suppose that M is the number of operating conditions. Given the sequence of sensor outputs, $o_1, \cdots, o_t$ up to time t, the *Viterbi* algorithm finds the most probable sequence of underlying states by solving the likelihood of the given sensor outputs. That is, the algorithm chooses the most likely state which produces the maximum likelihood value. Mathematically, the algorithm proceeds as follows:

1. *Initialization*: Set $\alpha_1(j) = \dot{\pi}_{j0} SE_j(o_1)$, $\forall j \in \{1, 2, \cdots, M\}$.
2. *Recursion*: For, $2 \leq n \leq t$ compute $\alpha_n(j) = max_i\{\alpha_{n-1}(i) p_{ij} SE_j(o_n)\}$, $\forall j \in \{1, 2, \cdots, M\}$.

where $\dot{\pi}_{j0}$ is the estimated probability that the initial system state is j. Then, the estimated current state $\hat{Z}_t$ is given by

$$\hat{Z}_t = \underset{j \in \{1,2,\cdots,M\}}{\text{argmax}} \{\alpha_t(j)\}. \tag{6.10}$$

VI.1.6. Smart sensor model

In general, the above-described sensors are relatively cheap but may be unreliable. In practice, there are different means to evaluate the exact deterioration level of a component of a system. The most common way is to dispatch a maintenance

crew to conduct an on-site inspection. With emerging of new sensor technology, more advanced sensors (e.g., high-speed imaging or mobile acoustic sensors) can be invoked to do a follow-up investigation upon the initial sounding of alerts or alarms. Regardless of which mechanism is used, the follow-up investigation is usually much more expensive (and more accurate). Collectively, we refer to this observation mode as "Smart Sensor". We assume that once a "Smart Sensor" is invoked, the system state can be revealed with certainty.

VI.1.7. Maintenance model

We consider two different maintenance approaches in our simulation model: SchM and CBM. The SchM model reflects current maintenance practices. Wind farm operators usually carry out SchM twice a year in the low wind speed seasons (Pacot *et al.*, 2003). Following the industry practice, we set the major preventive repairs to be performed in spring and fall seasons in our simulation model.

The second model is a CBM model. Modern wind turbines are equipped with automated alarm call-out systems inside condition monitoring equipment so that when a sensor signal exceeds a certain threshold, an alarm is sent to a wind farm operator by fax, pager, simple message service (SMS) or email (Pacot *et al.*, 2003). In our simulation model, when the most probable state of a component, decided from the state evaluation module, is the *alarm* state, the alarm message is sent to the operation center. Then, we invoke *smart sensors* to evaluate the system conditions exactly. If the actual system state that the smart sensors report is equal to the *alarm* state, PM has to be carried out.

In both maintenance strategies, unplanned failures are not completely avoidable. When a wind turbine fails before SchM or PM, we carry out corrective maintenance. In this case, we consider a substantial lead time for organizing maintenance crews

and spare parts.

As discussed earlier, maintenance actions can be constrained by adverse weather conditions (McMillan and Ault, 2008, Pacot *et al.*, 2003). Therefore, repair actions cannot be carried out under these harsh weather conditions. We reflect these kinds of weather constraints in the simulation model. In our simulation model, we assume that maintenance actions are carried out only when wind speeds are less than 10 m/s.

VI.2. Performance measures

We consider four performance measures to evaluate wind farm operations and maintenance.

Capacity factor. Capacity factor quantifies the productivity of a wind turbine (American Wind Energy Association, 2008). It compares the turbine's (or wind farm's) actual production over a given period of time with the amount of power the turbine (or wind farm) would have produced if it had run at full capacity for the same amount of time. Let P_{actual} denote the actual amount of power produced over a given time period and let P_{ideal} denote the amount of power that would have been produced at full capacity. Then capacity factor can be expressed as the ratio $\frac{P_{\text{actual}}}{P_{\text{ideal}}}$.

Availability. Availability refers to the percentage of time a wind turbine is available to generate power and is not out of service or under repair.

Number of failures of a given period. We count the number of times a wind turbine fails over a given period. This is an important factor for measuring the reliability of a wind turbine.

Wind farm O&M costs. We collect maintenance costs over a given time period to quantify the effectiveness of the different maintenance strategies.

VI.3. DEVS preliminaries

Before presenting the DEVS wind farm simulation model, we first provide some preliminaries on DEVS.

Parallel DEVS (Zeigler *et al.*, 2000) follows a hierarchical approach to building complex models from the basic models called *atomic* models. DEVS has a well defined concept of component coupling. Atomic models are coupled to form *coupled* (composite) models. In DEVS coupled models are treated as components via the property of *closure under coupling*, which enables the hierarchical model composition construct. An atomic model possesses *input* and *output ports* through which all interaction with the environment is mediated and has to be in a defined state at any given time. The input ports allow the atomic model to receive external *events* arising outside the model. Coupling is done by joining the output of one atomic model to the input of another to enable communication between models. The description of the internal workings of the model determines how the model responds to external events. The internal events arising within the model can change its state and manifest themselves as events on the output ports to be transmitted to other models. Parallel DEVS allows for all imminent atomic models to be activated and send their outputs to other components of the system.

Next we provide the mathematical description of a Parallel DEVS model. Let us denote by M an atomic model with a set of input ports $IPorts$, a set of input values (events) X_p, a set of output ports $OPorts$, and a set of output values (events) Y_p. Let (p, v) denote the port-value pair. Then a basic *Parallel DEVS* is a structure defined as follows:

Definition 5.

$$DEVS = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda_{out}, ta)$$

where,

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ *is the set of input ports and values, where* $IPorts$ *is the set of input ports;*

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ *is the set of output ports and values;*

S *is the set of sequential states;*

$\delta_{ext} : Q \times X_M^b \rightarrow S$ *is the* external transition *function, where* X_M^b *is a set of* bags *over elements in* X_M *and* Q *is the set of total states;*

$\delta_{int} : S \rightarrow S$ *is the* internal state transition *function;*

$\delta_{con} : Q \times X_M^b \rightarrow S$ *is the* confluent transition *function;*

$\lambda_{out} : S \rightarrow Y_M^b$ *is the output function;*

$ta : S \rightarrow R_{0,\infty}^+$ *is the* time advance *function; and*

$Q := \{(s, e) | s \in S, 0 \leq e \leq ta(s)\}$ *is the set of total states, where* s *is the state and* e *is the elapsed time.*

Parallel DEVS has the ability to handle multiple inputs and uses a *bag* to store the inputs. A bag is a set with possible multiple occurrences of its elements.

According to Definition 5, at any time the system is in some state s and if no external events occur, the system will remain in its current state for a time $ta(s) \in [0, \infty]$. When this time expires the system outputs the value, $\lambda_{out}(s)$, and transitions to a state $s' = \delta_{int}(s)$. In DEVS an output is only possible after an internal transition. If an external event $x \in X_M$ occurs when the system is the total state (s, e) with $e \leq ta(s)$, the system changes to state $s' = \delta_{ext}(s, e, x)$. The external transition function determines a new state when an external event occurs while the internal transition

function determines a new state when no events occur since the last transition. The confluent function decides the next state in cases when there is an external event exactly when an internal transition has to occur.

To construct models from components (DEVS models), the DEVS specification includes the external interface, the components, and the coupling relations. Let EIC, EOC and IC denote the external input coupling, external output coupling and internal coupling, respectively. Then a coupled model N can be defined mathematically as follows:

Definition 6.

$$N = (X, Y, D, \{M_d \mid d \in D\}, EIC, EOC, IC)$$

where,

$$X = \{(p, v) | p \in IPorts, v \in X_p\}$$

is the set of input ports and values and

$$Y = \{(p, v) | p \in OPorts, v \in Y_p\}$$

is the set of output ports and values. D is the set of component names, and for each $d \in D$,

$$M_d = (X_d, Y_d, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda_{out}, ta)$$

is a DEVS model with

$$X_d = \{(p, v) | p \in IPorts_d, v \in X_p\}$$

and

$$Y_d = \{(p, v) | p \in OPorts_d, v \in Y_p\}.$$

The external input coupling, EIC, connect external inputs to component inputs:

$$EIC \subseteq \{((N, ip_N), (d, ip_d)) \mid ip_N \in IPorts, d \in D, ip_d \in IPorts_d\}.$$

The external output coupling, EOC, connect external outputs to component outputs:

$$EOC \subseteq \{((N, op_d), (N, op_N)) \mid op_N \in OPorts, d \in D, op_d \in OPorts_d\}.$$

Lastly, the internal coupling, IC, connect component outputs to component inputs:

$$IC \subseteq \{((a, op_a), (b, ip_b)) \mid a, b \in D, op_a \in OPorts_a, ip_b \in IPorts_b\}.$$

Finally, DEVS does *not* allow for an output port of a component to be connected to an input port of the same component. Thus in DEVS $(a, op_a), (b, ip_b) \in IC$ implies $a \neq b$. In other words, no direct feedback loops are allowed for each component.

VI.4. Atomic models

Let us now turn to deriving DEVS atomic and coupled models for the DEVS wind farm simulation. These models provide the basic building blocks for the overall wind farm simulation model. We follow a bottom-up approach by first deriving atomic models, coupling them to create coupled models, and then coupling the coupled models to create to the overall simulation model. In this section, we describe the atomic models and the next section will discuss how we couple the atomic models to make a generic simulation model for wind farm operations.

The DEVS wind farm simulation model we propose comprises the models abstracted in Section VI.1. Specifically, we derive the following atomic models: Power Generator (PWRGEN), Component Degradation (CMPDEG), Wind Generator (WGENR), Sensor (SENSR), Smart Sensor (SMSENSR), State Evaluation (STEVAL),

Maintenance Scheduler (MSCHEDR), Maintenance Generator (MGENR), and Transducer (TRANSD). PWRGEN models electrical power generation based on wind speeds from the WGENR atomic model. The amount of power generated at any given time is calculated using a power curve in equation (6.1). The WGENR atomic model generates wind speeds calculated using the spatio-temporal wind model described in equations (6.5)-(6.9). CMPDEG models the degradation or deterioration process of wind turbine component.

SENSR and SMSENSR models unreliable sensor and smart sensor behavior, respectively, as described in Section VI.1. The STEVAL atomic model is responsible for estimating the actual state of a wind turbine gearbox based on the sensor information from SENSR and/or SMSENSR.

MSCHEDR models the two types of maintenance strategies, SchM and CBM, described in Section VI.1. The MGENR atomic model is responsible for generating the actual maintenance actions (e.g. dispatching a maintenance crew) based on the maintenance schedules created by MSCHEDR. Finally, TRANSD is in charge of computing the performance measures and statistical parameters of interest during the simulation run.

In this section, we present the details on two atomic models, PWRGEN and CMPDEG, to illustrate the derivation of the atomic models. We put the mathematical expressions of the two atomic models in Parallel DEVS in the Appendix A. The detailed descriptions of the rest of the atomic models (sensor (SENSR) model, smart sensor (SMSENSR) model, state evaluation (STEVAL) model and maintenance scheduler (MSCHEDR)) are also presented in Appendix A for the interested reader for testing and verification purposes.

VI.4.1. Power generator (PWRGEN) atomic model

The power generator (PWRGEN) atomic model has several basic states; *off_normal*, *off_normal_waiting*, *on_normal*, *off_alert*, *off_alert_waiting*, *on_alert*, *off_alarm*, *off_alarm_waiting*, *on_alarm*, *failed*, and *report_status*. We consider a PWRGEN with the input and output ports as shown in Fig. 19. The model has seven basic input ports, namely; 'turb_on_off", "wind_in", "deg_in", "corr_mnt", "prev_mnt", "obsv", and "req_status".

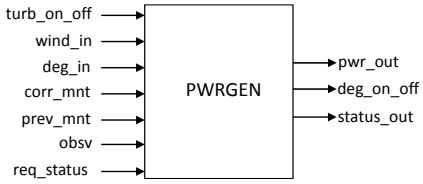


Fig. 19. Power generator (PWRGEN) atomic model

The operation of the PWRGEN atomic model is depicted in Fig. 20. The model is initialized in the *off_normal* state. When an input is received on the "turb_on_off" input port, the wind speed at the current location is verified. A transition to the *on_normal* state occurs if the current wind speed is within the thresholds specified for the proper operation of the turbine. If the current wind speed is outside the thresholds, the model transitions to *off_normal_waiting* state. Messages received on the "wind_in" input port will notify changes in the wind speed. A transition from *off_normal_waiting* state to *on_normal* state occurs when the wind speed satisfies the thresholds required for wind turbine operation. If an input is received on the "prev_mnt" input port or on the "turb_on_off" input port when the model is in *on_normal* state, a transition to *off_normal* state occurs.

A transition from *on_normal* state to *on_alert* state occurs if a message is received on the "deg_in" input port containing the information of a component degradation

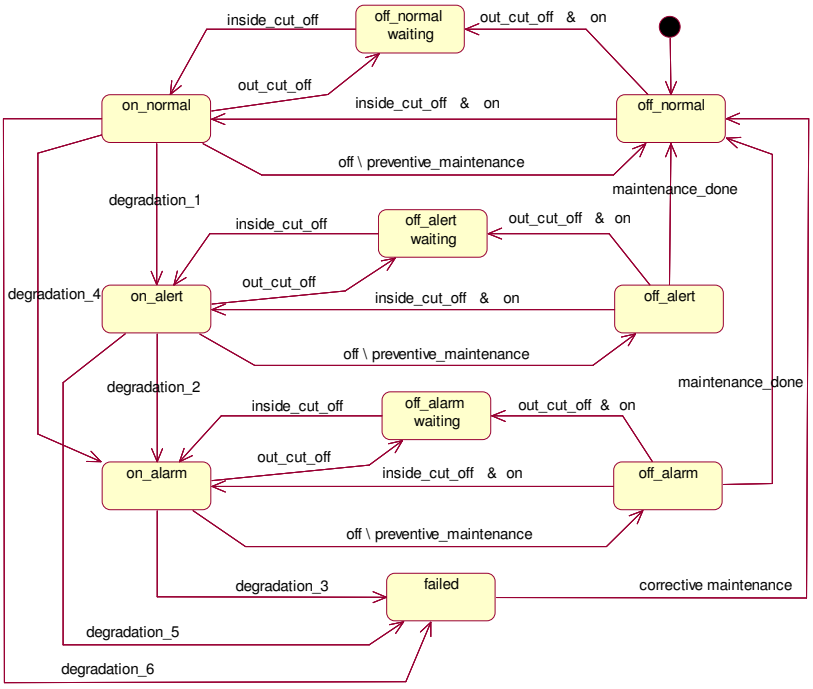


Fig. 20. Power generator (PWRGEN) state transition diagram

of *alert* (type 1). When in *on_alert* state, the model transitions to *off_alert_waiting* state if the wind speed goes out of the specified thresholds, or to *off_alert* state if the component is turned off or if PM has to be performed. In the case PM is performed, the model transitions from *off_alert* to *off_normal* state upon completion of maintenance.

A transition to *on_alarm* happens if the model is in *on_normal* or *on_alert* state and a message is received on the "deg_in" input port containing the information of a component degradation of *alarm* (type 2 or type 4). When the model transitions to *off_alarm_waiting* state if the wind speed goes out of the specified thresholds, or to transitions to *off_alarm* state if the component is turned off or if PM has to be performed. In the case PM is performed, the model transitions from *off_alarm* state

to *off_normal* state upon completion of maintenance.

A transition to *failed* state occurs if the model is in one of the following states; *on_normal*, *on_alert* or *on_alarm*; and a message is received on the "deg_in" input port containing the information of a component degradation of *failure* (type 3, type 5 or type 6). The model transitions from *failed* state to *off_normal* when a message is received on the "corr_mnt" input port indicating that the turbine has been fixed after corrective maintenance has been performed. When a message is received on the "req_status" input port, the model transitions to *report_status* state and remains in this state for a short time interval (STI). Once this STI has elapsed, the model transitions to the original state it was before receiving the message.

The PWRGEN atomic model has three output ports, namely; "pwr_out", "deg_on _off", and "status_out". The 'pwr_out" output port is used to notify the amount of power generated by the turbine during a period of time. A message is sent using the "deg_on_off" output port when the turbine is turned off. This message will turn off other components that are linked to this model. The "status_out" output port is used to report the status of the turbine when requested by the smart sensor. A mathematical expression of the PWRGEN atomic model in Parallel DEVS is given in the Appendix.

VI.4.2. Component degradation (CMPDEG) atomic model

CMPDEG is a model of a critical component of a wind turbine and in our case, represents the degradation of a wind turbine gearbox. This atomic model has six basic states; *passive*, *active*, *passive_wind*, *report_status*, *report_deg*, and *passive_service*. CMPDEG is always coupled to a PWRGEN atomic model to create the wind turbine (WTURBINE) coupled model. The coupling between these two models is

discussed in Section VI.5.1. CMPDEG has three input ports, namely; "wind_on_off", "main_on_off", and "manual_on_off". Fig. 21 shows the input and output ports for the model, while its operation is depicted in Fig. 22.

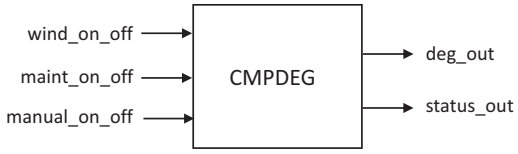


Fig. 21. Component degradation (CMPDEG) atomic model

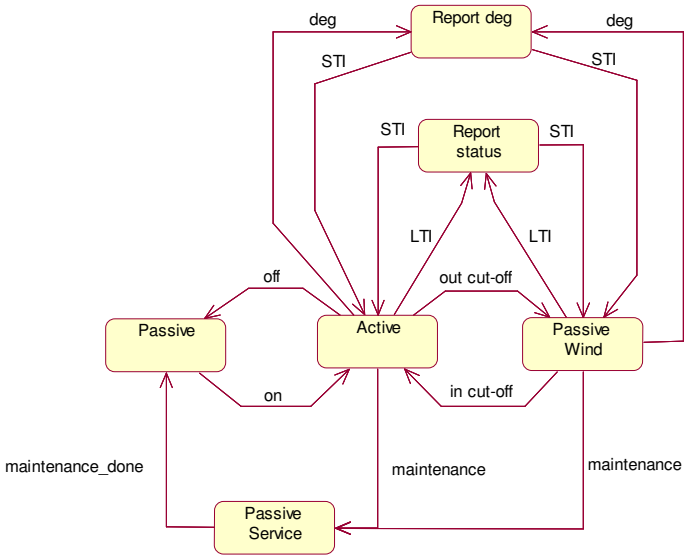


Fig. 22. Component degradation (CMPDEG) state transition diagram

CMPDEG is initialized in *passive* state. If an input is received on the "manual_on_off" input port, the model transitions to *active* state. Once in *active* state, three things can happen. First, a message received at the "wind_on_off" input port will indicate that the component has to be turned off due to current wind speed being above the threshold. In this case the model transitions to *passive_wind* state. Second, a transition to the *report_status* state occurs when a predetermined large

time interval (LTI) has elapsed. The *report_status* state is used to report the current component status to the sensors. Third, a change in the status of the model due to degradation will take the model to *report_deg* state. The model will transition to the *passive_service* state if it is in *active* or *passive_wind* state and a message is received on the "maint_on_off" port. When maintenance is completed, the model returns to its initial *passive* state.

CMPDEG model has two output ports, namely; "deg_out" and "status_out". The first outport is used to send messages to the PWRGEN atomic model when a status change occurs. The second output port is used to report the current state of the component to the sensors. A mathematical expression of the CMPDEG atomic model in Parallel DEVS is given in the Appendix A.

VI.5. Coupled models

We couple the four atomic models, PWRGEN, CMPDEG, SENSR and STEVAL to create a Wind Turbine (WTURBINE) coupled model. A collection of several WTURBINE coupled models forms the Wind Farm (WF) coupled model. We couple MSCHEDR and MGENR to create an Operation and Maintenance (OPMNT) coupled model. Similarly, we couple WGENR and TRANSD to create an Experimental Frame (EF). Finally, we couple the three coupled models, WF, OPMNT and EF to form the DEVS wind farm simulation model. We provide descriptions of the three coupled models WF, OPMNT and EF using block diagrams. We end this section with a system entity structure (SES) to provide a summary of the hierarchical structure and possible structures of the DEVS wind farm simulation model.

VI.5.1. Wind turbine (WTURBINE) coupled model

WTURBINE is a coupled model comprising PWRGEN and CMPDEG atomic models. Fig. 23 gives the block diagram for the WTURBINE coupled model showing the input and output ports. Five input ports are defined for this coupled model, namely; "on_off", "wind_speed", "observation", "corrective_maintenance", and "preventive_maintenance". The atomic models communicate using two internal couplings (ICs). Information is passed to the CMPDEG model when the PWRGEN is turned off (on). Every time the PWRGEN atomic model is turned off, the CMPDEG atomic model is turned off as well. The CMPDEG atomic model notifies the PWRGEN atomic model about changes in the component's degradation status. Recall that if degradation occurs, PWRGEN transitions from the current state to a more degraded state, e.g. *normal* to *alert* state. The three output ports in Fig. 23, namely; "pwrgen_status_out", "comp_status_out", and "power_out" are used to report the status and performance of WTURBINE.

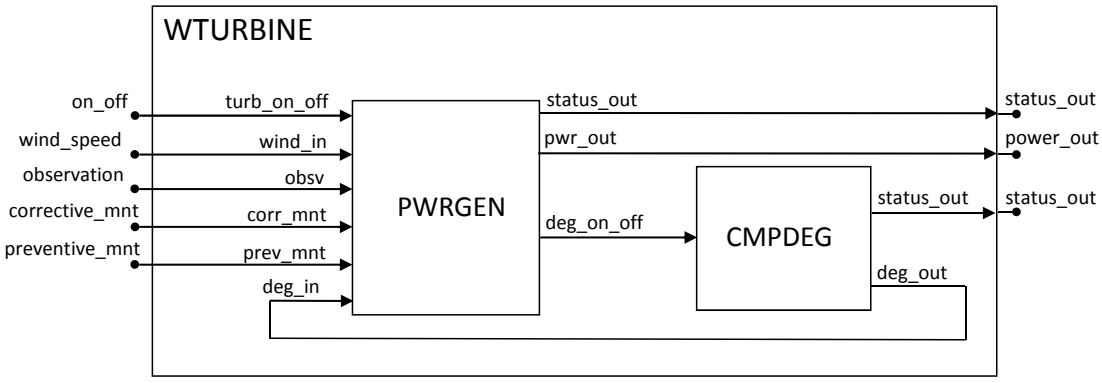


Fig. 23. Wind turbine block diagram with input and output ports

VI.5.2. Operation and maintenance (OPMNT) coupled model

OPMNT is a coupled model formed by coupling MSCHEDR and MGENR atomic models. Fig. 24 shows the block diagram for a OPMNT coupled model. One input port is defined for this coupled model, namely; "status_in". The MSCHEDR atomic model communicates with the MGENR atomic model by using one *IC*. Information is passed to the MGENR atomic model when a maintenance procedure is scheduled. MGENR uses the information to generate maintenance jobs at the scheduled times.

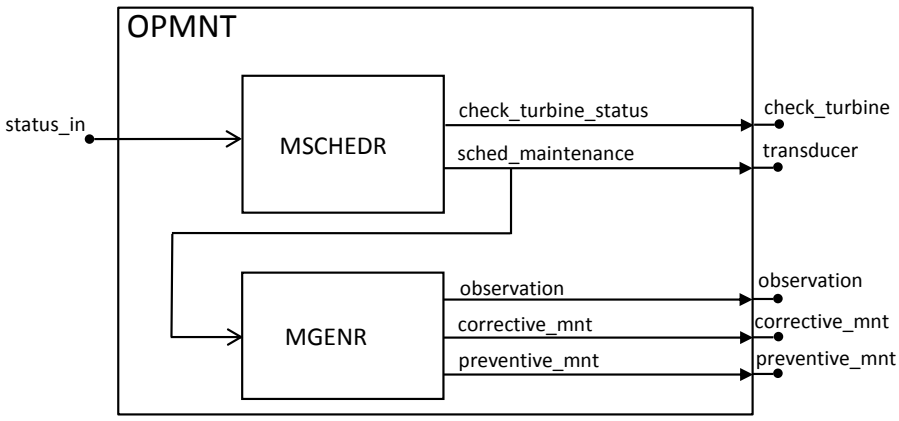


Fig. 24. O&M block diagram with input and output ports

VI.5.3. Experimental frame (EF)

The experimental frame is one of the most important components of the simulation model because it is used to define the experiment parameters and to collect the information of interest from the simulation runs.

In our wind farm model, EF is a coupled model that is formed by coupling WGENR and TRANSD atomic models. Fig. 25 shows the atomic models that are part of the EF coupled model and the way they are connected. Recall that the

WGENR atomic model is in charge of generating wind speed information for each one of the turbines used in the simulation. This model computes the wind speed for each WTURBINE atomic model based on its height and location in the wind farm. The TRANSD coupled model collects the information of interest and computes the performance measures specified by the user.

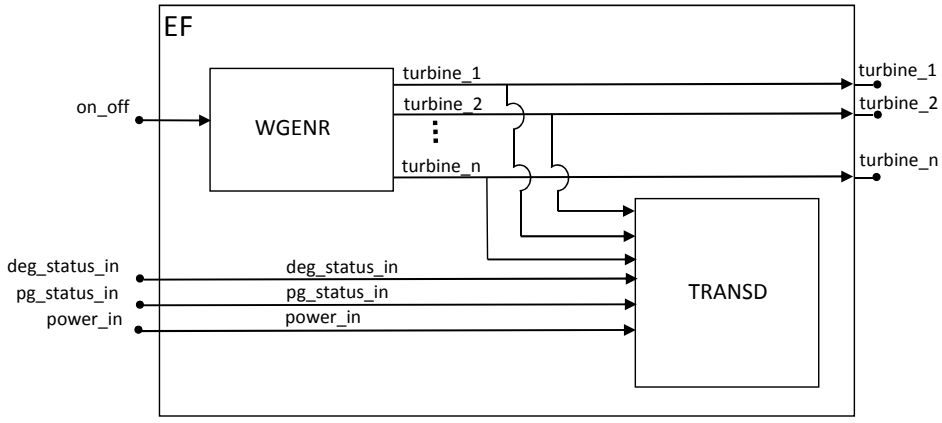


Fig. 25. Experimental frame (EF) coupled model

VI.5.4. Overall simulation model

The overall wind farm simulation model is depicted in Fig. 26. Due to space restrictions, the figure shows only a few wind turbines. In general, the wind farms (WF) would have several wind turbines coupled to the EF and OPMNT models. Observe that the EF and OPMNT are designed to be separated from WF. This allows for changes in the experimenter's goals or changes in the operations and maintenance policies to be done independently without concern for making changes to the WF coupled model.

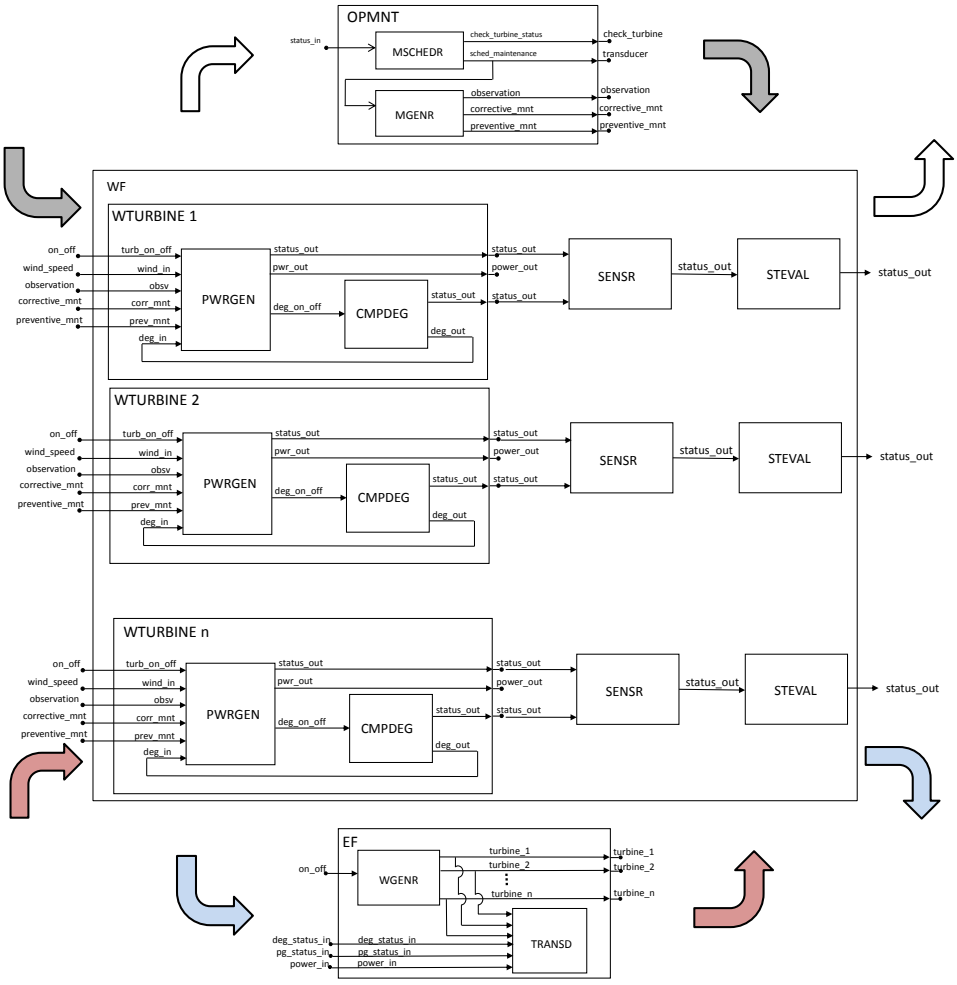


Fig. 26. DEVS wind farm system

VI.5.5. System entity structure

The system entity structure (SES) is utilized to plan, generate and evaluate design of simulation-based systems (Zeigler *et al.*, 2000). This is a scheme that organizes a set of possible structures of a system. A library of models is generated when all the components abstracted from the real system are implemented. The SES is used to classify these components by their characteristics and to organize them in hierarchical composition. This representation allows the modeler to visualize the system as a whole. The goal of the SES is to synthesize a simulation model by traversing a model

hierarchical structure. A SES represents not a single model structure, but a family of structures from which a candidate entity structure can be selected.

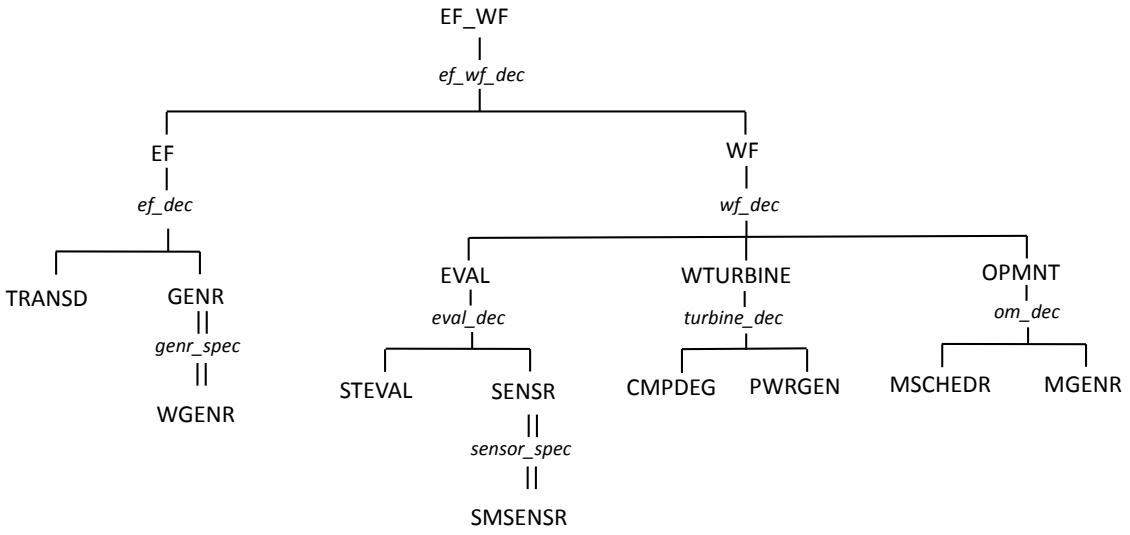


Fig. 27. System entity structure (SES) for the DEVS wind farm simulation

Fig. 27 shows the SES for the wind farm discrete event simulation. At the top level, the scheme shows the two major coupled models that define the system structure. The *Experimental Frame* (EF) branch can be decomposed into two branches that are assigned to the *Transducer* (TRANSD) and *Generator* (GENR) atomic models. The double line under the GENR branch means specialization. The GENR model can be categorized into a specialized entity called *Wind Generator* (WGENR). The *Wind Farm* (WF) branch can be decomposed into three branches: *System Evaluation* (EVAL), *Wind Turbine* (WTURBINE), and *Operation and Maintenance* (OPMNT). The EVAL branch can be decomposed into two branches, *State Evaluation* (STEVAL) and *Sensor* (SENSR). The SENSR model can be categorized into a specialized entity called *Smart Sensor*(SMSENSR). The *Wind turbine* (WTURBINE) and the *Operation and Maintenance* (OPMNT) branches are decomposed into the models that are used to form their respective coupled models.

VI.6. Application

The wind farm simulation model is implemented in DEVSJAVA (Zeigler and Sarjoughian, 2003) which is a Java-based software implementation of the DEVS formalism. All computational experiments are conducted on a DELL Optiplex GX620 with a Pentium D processor running at 3.0GHz with 3.5GB RAM.

The simulation model is verified and tested using DEVSJAVA SimView version 1.0.4 (Zeigler and Sarjoughian, 2003). This visual interface allows the modeler to inspect the behavior of each atomic and coupled model created in DEVSJAVA. Atomic models are inspected first because they should perform as expected to achieve the proper functioning of the coupled models. SimView has several convenient functionalities such as allowing the user to control the simulation run (start and stop), simulation fast-forwarding or slow motion, and being able to insert user defined parameters created for model verification and testing by using the models' input ports. The SimView user interface also provides a top menu that allows the user to select the appropriate model and run it with the click of a button. A simulation clock is always displayed on the interface during the simulation run as well as statistics of the active models by simply positioning the mouse cursor on top of the model block.

VI.6.1. Computational experiments design

We applied our simulation model to a 100-unit wind farm located in West Texas. The wind farm is assumed to operate 24 hours a day for 365 days a year. Several computational experiments are performed to validate the model and gain management insights into the impact of maintenance scheduling policies on system performance. Ten replications for each simulation run are made and a scheduling time horizon of 20 years is used since the average lifespan of a wind turbine is 20 years.

A preliminary validation of the simulation model is achieved by configuring the simulation as described based on the following parameter setting (Note that the parameters in the simulation model are different from the ones used in the case study in Chapter V).

Wind turbine configuration. We assumed all the wind turbines to be *GE sle*'s with 1.5 MW rated power with a hub height of 100 m. The parameters corresponding to *GE sle* wind turbines are listed in Table 7.

Table 7. GE 1.5 sle turbine specifications

cut-in speed	3.5 m/s
rated speed	14 m/s
cut-out speed	25 m/s
rated power	1.5 MW

Gearbox degradation. We consider one week as a transition period and use the following P matrix to represent a weekly-based deterioration process.

$$P = \begin{bmatrix} 0.95 & 0.04 & 0.01 & 0.00 \\ 0.00 & 0.97 & 0.02 & 0.01 \\ 0.00 & 0.00 & 0.94 & 0.06 \\ 0.00 & 0.00 & 0.00 & 1.00 \end{bmatrix} \tag{6.11}$$

Weather data. The parameters in the spatio-temporal model (6.7) are estimated using historical wind speed data from the West Texas Mesonet (West Texas Mesonet, 2008). The West Texas Mesonet is a network of meteorological monitoring instruments, dispersed across West Texas. Based on the wind characteristics at this wind farm location, a value of 0.31 for α in equation (6.9) is used, which is a typical value

for suburban areas (Gipe, 2000).

O&M practices. We simulate wind farm operations under the two maintenance strategies: SchM and CBM. As mentioned in Section VI.1, we assume that under SchM, maintenance actions are performed twice a year in low windy conditions regardless of the deterioration status of a wind turbine. Under CBM strategy, PM actions are carried out only when sensors in condition monitoring equipment produce *alarm* signals. We assume that preventive repairs in SchM and CBM for each wind turbine takes two days at a cost of $8,632 based on a study by Rademakers *et al.* (2003a). Under both SchM and CBM, CM is performed upon an unplanned failure. In this case, we consider a lead time of six weeks before repairing the turbine. The cost for performing corrective maintenance on a single gearbox is assumed to be $17,264. We also set the cost to invoke smart sensors to be about 10% of the corrective maintenance costs according to the suggestions of our industry partners. (The monetary unit of each cost factor in this example is originally in euros (Rademakers *et al.*, 2003a), but we converted from euros to the US dollar with an exchange rate of 1 euro = 1.3572 dollar). The number of available repair crews is fixed at five throughout the simulation.

VI.6.2. Simulation results and discussion

The simulation results for the average total power generation and the capacity factor for 20 years under SchM and CBM, respectively, are reported in Table 8. We also report the simulation computational time. The columns of the table show the mean and standard deviation (in parenthesis) for each scheduling strategy. The results show that CBM performs better than SchM on both power generation and capacity factor. There is an increase of about 5.85 % (70,3968 MW) in power generation and increase of about 5.92 % in capacity factor under CBM. Each simulation run took

about 1.2 hours due to the large number of wind turbines (100) and the lengthy planning horizon (20 years) used.

Table 8. Simulation results for power generation and capacity factor (standard deviation in parenthesis)

	SchM	CBM
Power Generated (MW)	12,025,091.98 (18,851.97)	12,729,060.05 (10,488.85)
Capacity Factor	0.422 (0.001)	0.447 (0.001)
CPU Time (secs)	5,102.67 (84.23)	5,023.02 (7.79)

Annual power generation for each maintenance policy is reported in Fig. 28. The results show that CBM outperforms SchM for all the years except in year 12, where SchM has a slightly higher power output. Also, observe that the amount of power generation decreases annually under SchM starting in year 14. This is the time when the wind turbines are near their lifespan and are more prone to failure. The results indicate that there is relatively more corrective maintenance under SchM than CBM.

We also report on the 'accumulated' capacity factor, i.e. the average capacity factor based on the number of years from the start of the simulation. The results are plotted in Fig. 29 and show steady capacity factors for both SchM and CBM. However, CBM has relatively larger capacity factor than SchM. This means that on average, relatively more power is generated under CBM throughout the wind farm operation years. The capacity factor in actual wind farms is between 0.25 and 0.4 (American Wind Energy Association, 2008), which we believe is typically reported for SchM since SchM is the most widely used maintenance policy in the wind power

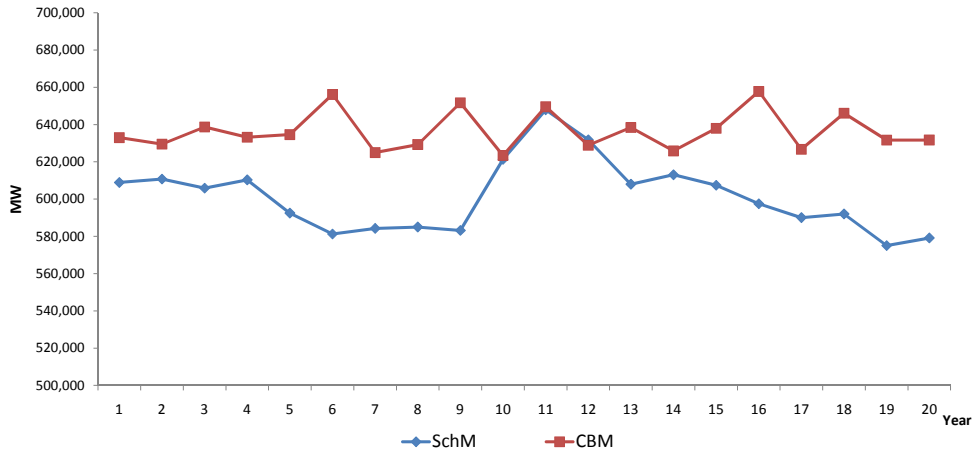


Fig. 28. Annual power generation

industry. In our case we obtain slightly higher values for capacity factor. We believe this is due to the fact that we only consider gearbox failures. Incorporating failures of other wind turbine components such as blades and generator, the capacity factor under SchM would fall within the range reported in the literature. Furthermore, the capacity factor under CBM is expected to be higher since CBM outperforms SchM.

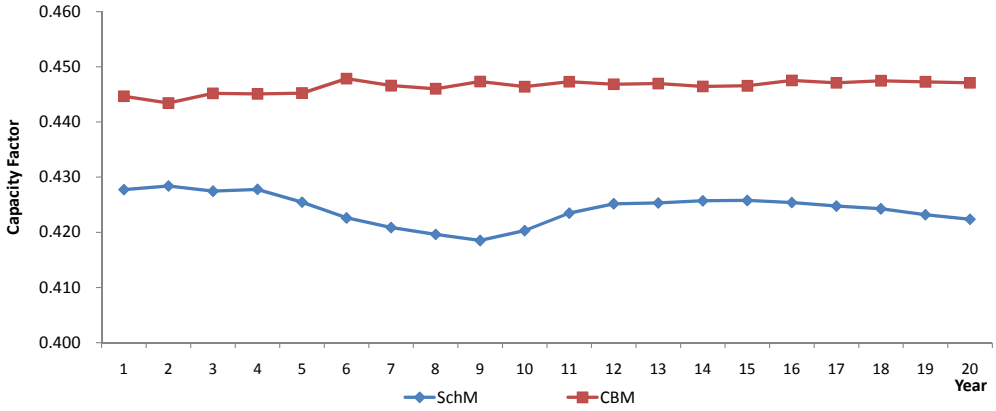


Fig. 29. Accumulated average capacity factor

The rest of the performance results are reported in Table 9. The table shows the average values of wind turbine availability, number of failures per turbine, and O&M costs per turbine over the 20-year period.

Table 9. Performance results (standard deviation in parenthesis)

	SchM	CBM
Availability	0.898 (0.008)	0.907 (0.006)
Number of failures per turbine per year	0.932 (0.014)	0.823 (0.011)
O&M costs per turbine per year ($)	33,364.47 (233.44)	22,839.72 (186.68)

The results show that CBM has about 1.0 % higher wind turbine availability over SchM. CBM also has about 11.7 % less number of gearbox failures than SchM. In terms of average total maintenance cost, SchM has a significantly higher cost (about 31.5% higher) than CBM. Recall that under SchM, PM is always performed twice a year on each turbine even when it may not be necessary, thus contributing to the higher O&M costs. Under CBM, PM is only performed when necessary based on sensor information. Even though CBM has much lower O&M costs, one has to also factor in the cost of managing and maintaining the sensors. Since we are unable find cost figures related to sensor management in the literature, we do not factor in such a cost into the total average maintenance cost.

We should point out that availability for onshore wind turbines reported in the literature is about 0.98 (Ribrant, 2006), which is higher than what we obtain (about 0.90). The low availability we obtain is due to the fact that we use a fixed and relatively long lead time of six weeks for CM of unplanned gearbox failures. According to a study by Ribrant (2006), the downtime upon a gearbox failure widely varies from

1 hour to 2,067 hours (almost 12 weeks). In our simulation we use the mid-range value of six weeks for lead time. However, in practice this time can vary significantly. We should also mention that the average number of failures per turbine in a year under the SchM policy, 0.932, falls within the range of actual gearbox failures, which is 0.05-2.29 times per year.

Fig. 30 compares the average number of failures per year during the 20-year period. The results show that the average number of failures is always higher under SchM strategy. Also, notice that there is a cyclic failure pattern similar to a sinusoid curve in both maintenance polices, which indicate the general Markovian degradation pattern followed by gearbox failures.

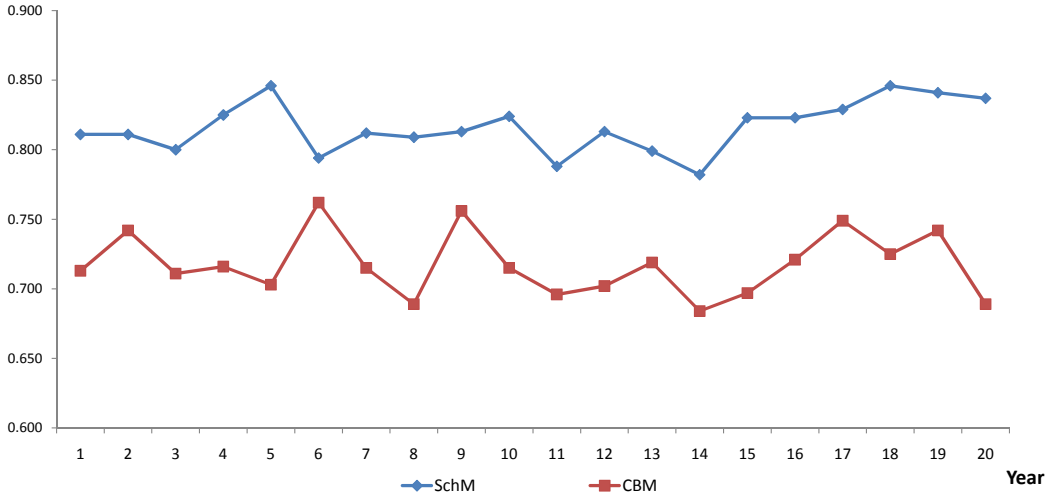


Fig. 30. Average number of failures per turbine

Fig. 31 depicts the 'accumulated' average number of failures at a given time period, i.e. the average number of failures from year one up to a given year. The results show that SchM has higher average number of failures at each time period. It is interesting to notice that there is an increasing number of failures in the last five years under the SchM policy. This can be attributed to the fact that we observe more failures than the available maintenance crews can handle towards the end of the wind

turbine lifespan. Recall that the number of repair crews is fixed at five throughout the simulation. Wind turbines that are close to failures during those years have to wait for repair crews to finish PM already scheduled for other turbines. The limited maintenance resources thus result in long downtime leading to increased revenue losses. On the contrary, the accumulated average number of failures under CBM remains steady because the CBM strategy utilizes maintenance resources only when repairs are needed to avoid failures. This result indicates that CBM is a beneficial maintenance strategy in wind farms with limited repair resources.

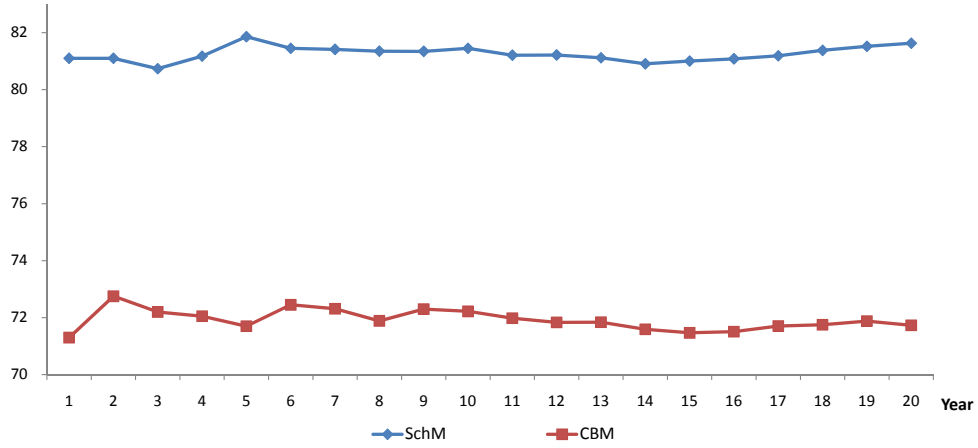


Fig. 31. Accumulated average number of failures per wind turbine

To assess the performance of each individual wind turbine, we plot the average availability (left axis) and the number of failures (right axis) for each wind turbine for the 20-year period under SchM and CBM in Fig. 32 and Fig. 33. The average availability and number of failures for SchM are between 0.87 - 0.92 and 15 - 22, respectively. Under CBM, the average availability and number of failures are between 0.89 - 0.93 and 13 - 20, respectively. Thus CBM has relatively better values for both availability and number of failures for each turbine. Finally, Fig. 34 plots the availability for each turbine for both SchM and CBM. As can be seen in the graph,

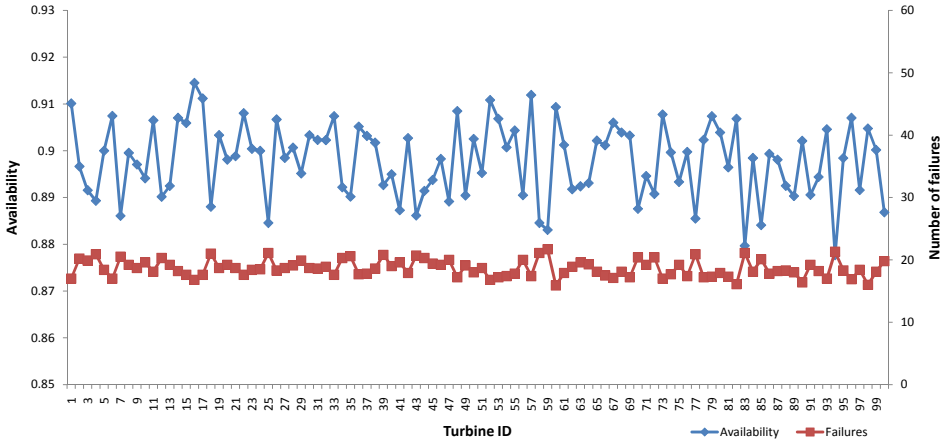


Fig. 32. Aailability and number of failures per wind turbine under SchM

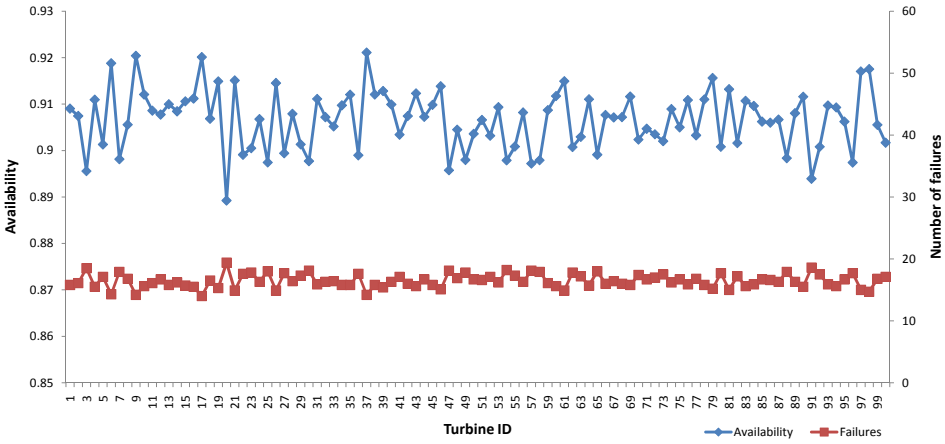


Fig. 33. Availability and number of failures per wind turbine under CBM

CBM gives higher availability for most of the wind turbines. Also, CBM has lesser variability in average availability among the wind turbines.

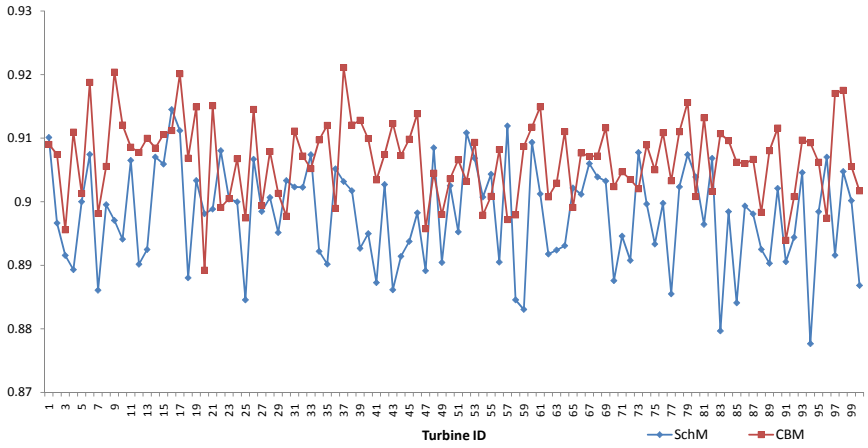


Fig. 34. Average availability per wind turbine under both scheduling algorithm

CHAPTER VII

THE INTEGRATION OF OPTIMIZATION MODELS IN THE SIMULATION FRAMEWORK

The optimization strategies, discussed in Chapters IV and V, are constructed with a set of assumptions which simplify some of the details of actual turbine operations. Consequently, there is a need to evaluate such strategies by use of the simulation model before field implementation. This chapter discusses our proposed integration framework regarding how to incorporate the optimization results into the simulation. We propose a real-time decision-making process for the dynamic CBM model based on the structural results garnered from the static CBM model, and describe the preliminary results.

VII.1. Real-time optimization algorithm

As discussed previously, our static CBM model contains a set of decision rules to determine the proper O&M action. Being static, it cannot adapt to the changing operating environments, but it is feasible for wind farms that operate in relatively stationary weather conditions. Applying the dynamic CBM model can provide more satisfactory results when there are strong seasonal variations, but it, too, has limitations. Its algorithm is not computationally efficient for large size problems with large planning horizon and/or with large state dimension (M; L). The heterogeneity of model parameters, by nature, does not allow us to easily derive structural properties which can save the computational efforts, unlike the static CBM model.

It is important to note that the two models and the resulting optimal strategies are based on *offline design*, i.e. the model parameters and decision-making procedure

are established offline using historical data yet real-time weather patterns will be quite different. For example, suppose that the current week is the tenth week of a year. In the dynamic CBM model we set $W_{CM(l),n} = 0.3$ for the current week based on historical data, and we cannot guarantee that the same probability occurs in the current week. In this case, the optimal policy based on historical data cannot adapt to these real-time, detailed weather conditions. Hence, an alternative strategy is a real-time optimization which utilizes refined weather patterns with these two features:

- The use of rule-based decision boundaries similar to those developed in the static CBM model.
- Coupling the most recent weather information for the current and a few next periods to the real-time decision boundaries.

The first feature reduces computational efforts considerably compared to the backward dynamic programming that solves the dynamic CBM model, and incorporating the optimization results is simple. Since it is rule-based and expressed as *if-then* rules, the strategy is easily implementable by operators and managers. The second feature fully utilizes refined, real-time weather information; thanks to the advent of today's sophisticated forecasting technology, accurate weather information is available for the current period and near future (The Weather Research and Forecasting (WRF), 2009). Having defined our decision rules, we can apply the most recent weather patterns to the rules at each decision point and decide the best action on-the-fly. The overall framework to incorporate the online decision-making approach in our simulation model is depicted in Fig. 35.

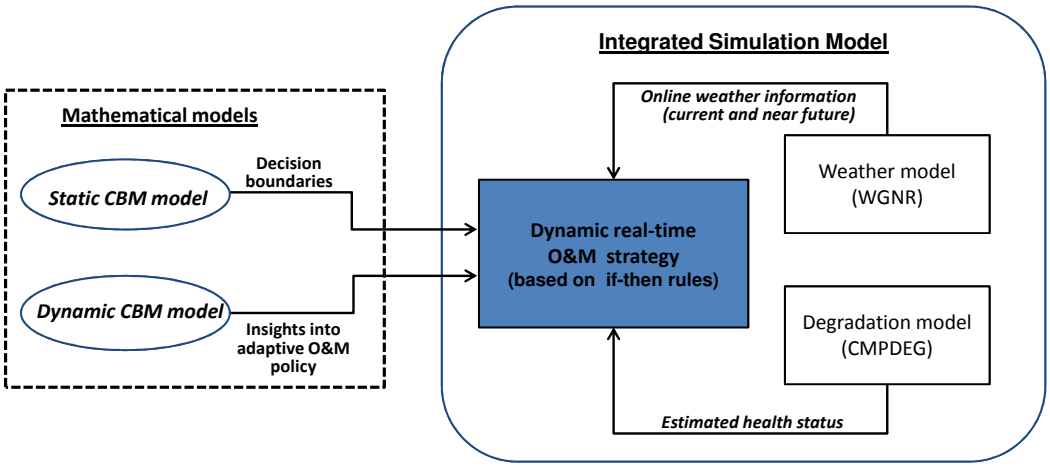


Fig. 35. Framework of real-time O&M strategy

VII.1.1. Extension of static CBM model with several failure modes and multiple preventive repair levels

Before developing the decision rules for our real-time O&M strategy, we briefly review the mathematical analysis discussed for the static CBM model and extend the model. Section IV.4 discussed how we derived a set of decision boundaries for each action to be optimal for the static CBM model with one failure mode and one preventive repair level, as summarized in Table 1. The following section generalizes the results with several failure modes and multiple preventive repair levels.

First, let us consider the $PM(m)$ actions with different repair levels, $m = 1, \cdots, M-1$. Recall that in our analysis in Section IV.4, we evaluate a bias associated with each action to determine the optimal policy. The bias associated with $PM(m)$, denoted by $b_{PM(m)}(\pi)$ in (4.15), is defined as follows:

$$b_{PM(m)}(\pi) = C_{PM(m)} + b(e_m) + \frac{(\tau - g)}{1 - W_{PM(m)}} \tag{7.1}$$

Since $b_{PM(m)}(\pi)$ is constant in π, we can define the best level of PM as m^* which gives the minimum bias among $(M-1)$ PM-associated biases. That is,

$$m^* \equiv argmin_{m=1,\cdots,M-1}\{b_{PM(m)}(\pi)\} \tag{7.2}$$

$$b_{PM(m^*)}(\pi) \equiv min\{b_{PM(m)}(\pi); m = 1, \cdots, M-1\} \tag{7.3}$$

Table 10 shows the conditions for each action to be optimal for the general static CBM model. In the table, $\delta^S(\pi)$ denotes the optimal policy at π. Since the claims in the table and their proofs are similar to the ones in Section IV.4, we omit them. Note that Table 1 in Section IV.4 is a special case when only one failure mode and one preventive repair level ($PM(1)$) is considered (i.e. $L = 1$ and $M = 1$).

VII.1.2. Modified decision rules for real-time decision making

In this section, we adjust the decision boundaries developed for the static CBM model in order to derive the real-time decision rules.

Recall that the decision boundaries for the static CBM model, summarized in Tables 1 or 10, are developed by evaluating the bias associated with each action, and the biases are computed based on the convergence property discussed in Section IV.2. That is, when the number of decision horizon goes to infinity, the optimal value function approaches to a linear line with a slope of the average O&M cost per period g, and an intercept $b(\pi)$ when the current information state is π (See (4.7) in Section IV.2).

According to Puterman (1994) (page # 338), the bias can be obtained during the first few transitions for an aperiodic Markov chain in an average cost Markov decision process model. Although we consider an infinite horizon model in the static CBM model, the decision horizon need not be very large to satisfy the convergence property.

Table 10. Closed boundaries for optimal policy for general static CBM model

$\delta^S(\pi)$	Conditions	Remark
$PM(m^*)$	if and only if $\sum_{l=1}^{L}(C'_{CM(l)} - C'_{PM(m^*)})H_l(\pi) - g > 0$ and $C'_{PM(m^*)} < C_{OB} + \sum b(e_i)\pi_i$ for $\pi \prec_{st} \pi'(\pi)$	Sufficient and necessary
OB	if $\sum_{l=1}^{L}(C'_{CM(l)} - C'_{PM(m^*)})H_l(\pi) - g > 0$ and $C'_{PM(m^*)} \geq C_{OB} + \sum b(e_i)\pi_i$ or, if $\sum_{l=1}^{L}(C'_{CM(l)} - C_{OB} - \sum b(e_i)\pi)H_l(\pi) + R(\pi)(\sum b(e_i)(\pi'_i(\pi) - \pi_i) - g) \geq 0$ and $\delta^S(\pi^2) = OB$	Sufficient
NA	if $\sum_{l=1}^{L}(C'_{CM(l)} - C'_{PM(m^*)})H_l(\pi) - g \leq 0$ and $C'_{PM(m^*)} < C_{OB} + \sum b(e_i)\pi_i$ or, if $\sum_{l=1}^{L}(C'_{CM(l)} - C'_{PM(m^*)})H_l(\pi) - g \leq 0$ and $\sum_{l=1}^{L}(C'_{CM(l)} - C_{OB} - \sum b(e_i)\pi)H_l(\pi) + R(\pi)(\sum b(e_i)(\pi'_i(\pi) - \pi_i) - g) < 0$	Sufficient

This gives an important implication for the real-time algorithm. In practice, the frequency of harsh weather events (i.e. $W_{CM(l),n}$, $W_{PM(m),n}$, $\forall l, m$) may happen more often during storm seasons or winter season. However, it would still be reasonable to assume that during a given season, harsh weather frequency is almost constant. Also, we consider that wind farm operators want to make a timely decision to avoid catastrophic failures. For example, a weekly decision would be of practical choice. Considering the weekly decision-making process, each season consists of multiple periods. When the remaining periods until the next season are not very small, we can still apply the convergence property to determine the decision boundaries.

In applying the decision boundaries derived in the static CBM model to the real-time decision making (or dynamic CBM model), we need a few more approximations. First, note that in the static CBM model, we do not consider a discount factor because we use an average cost model, whereas in dynamic CBM model, β is not necessarily *one*. However, when decisions are made frequently, a discount rate is close to *one*. Also, recall that in the static CBM model, we compared actions pairwise (See Lemma 2 through Lemma 4 in Section IV.4). For the comparison of "$PM(m)$ vs. NA" and "$PM(m)$ vs. OB", we derive necessary and sufficient conditions to define which action is preferred to the other action. However, when we compare NA with OB, we derive only the sufficient condition under which NA is preferred. We approximate that if the condition does not hold, OB is preferred.

Next, how can we bind the most updated forecasting information to the decision rules? Recall that in the static CBM model, we define the new repair costs as $C'_{CM(l)}$, $C'_{PM(m)}$ in (4.16) and (4.17), respectively, to compound weather effects, lead time and production losses, as follows:

$$C'_{CM(l)} = C_{CM(l)} + \lambda(l) \cdot (\tau - g) + \begin{cases} \frac{W_{CM(l)}(\tau-g)}{1-W_{CM(l)}} & \text{if } \mu(l) = 0 \\ \frac{\tau-g}{1-W_{CM(l)}} & \text{if } \mu(l) = 1 \end{cases} \tag{7.4}$$

$$C'_{PM(m)} = C_{PM(m)} + b(e_m) + \frac{\tau - g}{1 - W_{PM(m)}} \tag{7.5}$$

where $\mu(\cdot)$ is the number of repair periods required when the system fails with l^{th} failure mode. In (7.4), the second term $\lambda(l) \cdot (\tau - g)$ represents the revenue losses during the lead time. Since in the dynamic CBM model, the revenue losses depend on the period, we replace the second term with $\tilde{\tau}_n(l) - \lambda(l) \cdot g$, where $\tilde{\tau}_n(l)$ is the total revenue losses during the lead time as defined in (5.2) with $\beta = 1$. Also, in (7.4), the third term reflects the expected revenue losses during repair delay after the maintenance resources are ready. Thus, we replace τ and g in the third term with

$\tau_{\lambda(l)-n-1}$ and $W_{CM(l),\lambda(l)-n-1}$, respectively (note that n implies the remaining periods until the terminal period; thus, $\lambda(l)-n-1$ in (7.4) is the $(\lambda(l)+1)^{th}$ period after the current period n). In (7.5), since $PM(m)$ can be carried out during the current period as long as weather conditions permits the corresponding preventive repairs, we can use the τ_n and $W_{PM(m),n}$ for τ and g, respectively. In summary, we redefine the repair costs as follows:

$$C^R_{CM(l)} = C_{CM(l)} + \tilde{\tau}_n(l) - \lambda(l)\cdot g + \begin{cases} \frac{W_{CM(l),\lambda-n-1}(\tau_{\lambda-n-1}-g)}{1-W_{CM(l),\lambda(l)-n-1}} & \text{if } \mu(l)=0 \\ \frac{\tau_{\lambda-n-1}-g}{1-W_{CM(l),\lambda-n-1}} & \text{if } \mu(l)=1 \end{cases} \tag{7.6}$$

$$C^R_{PM(m)} = C_{PM(m)} + b(e_m) + \frac{\tau_n - g}{1-W_{PM(m),n}} \tag{7.7}$$

Here, superscript R reflects the real-time strategy. We can now substitute these redefined cost factors and weather parameters into the bias functions in (4.18) as follows:

$$b^R(\pi) = min \begin{cases} b^R_{NA}(\pi) = \sum_{l=1}^{L} C^R_{CM(l)} H_l(\pi) + b^R(\pi'(\pi))R(\pi) - g^R, \\ b^R_{PM(m)}(\pi) = C^R_{PM(m)}, m = 1,\cdots,M-1 \\ b^R_{OB}(\pi) = C_{OB} + \sum_{i=1}^{M} b^R(e_i)\pi_i \end{cases} \tag{7.8}$$

Then, we can apply the policy iteration (or value iteration) along the extreme sample paths to find the average cost and extreme biases, denoted by g^R and $b^R(e_i)$, $i = 1,\cdots,M$, respectively.

Finally, we can use the preference conditions between two actions to find the best policy, as summarized in Table 11. Note that the claims in the table generalize the results for the static CBM model with $L = 1$ and $M = 1$ which is discussed in Section IV.4.

Table 11. Preference conditions for real-time decision-making

Actions	Preference Conditions
$PM(m^*)$ vs. NA	$PM(m^*)$ is preferred to NA if $\sum_{l=1}^{L}(C^R_{CM(l)} - C^R_{PM(m^*)})H_l(\pi) - g^R > 0$ and vice versa
$PM(m^*)$ vs. OB	$PM(m^*)$ is preferred to OB if $C^R_{PM(m^*)} < C_{OB} + \sum b^R(e_i)\pi_i$ and vice versa
NA vs. OB	NA is preferred to OB if $\sum_{l=1}^{L}(C^R_{CM(l)} - C_{OB} - \sum b^R(e_i)\pi_i)H_l(\pi) + R(\pi)\left(\sum b^R(e_i)(\pi'_i(\pi) - \pi_i)\right) - g^R_n < 0$ and vice versa

VII.1.3. Adjustment for transition periods

In highly complex systems with hundreds of wind turbines, the decision rules presented in Table 11 will provide considerable computational savings over the dynamic programming algorithm (**Algorithm V.1**). Nevertheless, an important limitation of this approximation is that only the current and predicted weather information for the near future is used to make decisions. Therefore, if the weather conditions are updated at a high frequency or change dramatically in the consecutive periods, erratic performance and instability could arise. The example can be observed during transitional periods (known as "shoulder seasons") from harsh to mild seasons. In these transitional periods, the optimal policy will encourage operators to postpone preventive repairs until the mild weather arrives (see case study in Chapter V). However, the real-time decision boundaries may not capture these dramatic weather changes.

To adjust the dramatic weather changes during transition periods, we must add one more rule. Note that the average number of periods to complete preventive repairs is $\frac{1}{1-W_{PM(m),n}}$. Therefore, when the predicted weather condition during the next $\left[\frac{1}{1-W_{PM(m),n}}\right]$ periods is better than the current condition, we do not perform PM and take NA at the current period. Here, $[\cdot]$ denotes the closest integer. This rule compensates the limitation of the closed decision boundaries described in Table 11 when the remaining periods to the upcoming next season are few.

VII.1.4. Real-time algorithm

Now, let $\delta^R(\pi)$ denote the policy from the real-time algorithm at a state π. Suppose that the number of remaining periods to the terminal period is n. We summarize the approximate algorithm as follows:

Algorithm VII.1. Construction of a real-time algorithm

Input: $\lambda(l), \mu(l), C_{CM(l)}, C_{PM(m)}, C_{OB}, l = 1, \cdots, L, m = 1, \cdots, M-1$.

Step 1. *Based on the most recent weather forecasting information, estimate the weather related parameters,* τ_n, $\tau_{\lambda(l)-n-1}$, $W_{CM(l),\lambda(l)-n-1}$, *and* $W_{PM(m),n}, \cdots, W_{PM(m),n-K(m)}$ *where* $K(m) = \left[\frac{1}{1-W_{PM(m),n}}\right]$

Step 2. *Compute* $C^R_{CM(l)}$, $C^R_{PM(m)}$ *using (7.6) and (7.7), respectively*

Step 3. *Compute the biases at the extreme points,* $b^R(e_i)$, $i = 1, \cdots, M+L$ *and average cost* g^R_n *by applying the standard policy iteration (or value iteration) to the states along the extreme sample paths*

Step 4. *Then, apply the following decision rules:*

(a) *Find the most preferable* PM *action* $PM(m^*)$ *which gives the minimum* $C^R_{PM(m)}$ *in (7.7),* $m = 1, \cdots, M-1$

(b) *Suppose that* $\sum_{l=1}^{L}(C^R_{CM(l)}-C^R_{PM(m^*)})H_l(\pi)-g^R>0$. $\delta^R(\pi)=PM(m^*)$ *if* $C^R_{PM(m)}<C_{OB}+\sum_{i=1}^{M}b_n(e_i)\pi_i$. *Otherwise,* $\delta^R(\pi)=OB$

(c) *Suppose that* $\sum_{l=1}^{L}(C^R_{CM(l)}-C^R_{PM(m^*)})H_l(\pi)-g^R\leq 0$. $\delta^R(\pi)=NA$ *if* $\sum_{l=1}^{L}(C^R_{CM(l)}-C_{OB}-\sum b_n(e_i)\pi)H_l(\pi)+R(\pi)(\sum b_n(e_i)(\pi_i'(\pi)-\pi_i)-g^R)<0$. *Otherwise,* $\delta^R(\pi)=OB$

Step 5. *Take* NA *(i.e. set* $\delta^R(\pi)=NA$*) if* $\delta^R(\pi)=PM(m^*)$, *but* $W_{PM(m^*),n-k}<W_{PM(m^*),n}$ *for any* $k=1,\cdots,K(m^*)$

Remark. Note that in Step 3, the policy iteration (or value iteration) is applied *only* to extreme sample paths to obtain g^R and $b^R(e_i)$ values, leading to substantial reduction of computational time and complexity when compared with the dynamic program algorithm (**Algorithm V.1**). More approximation can be made by using fixed g^R and $b^R(e_i)$ values. The preliminary results using a wide range of weather parameters ($W_{CM(m),n}$,$W_{PM(m),n}$) show that these g^R and $b^R(e_i)$ values are not sensitive to the weather parameters. Consequently, as an alternative to Step 3, we can compute the g^R and $b^R(e_i)$ a priori using average values of $W_{CM(m),n}$,$W_{PM(m),n}$ and use the numbers as input parameters.

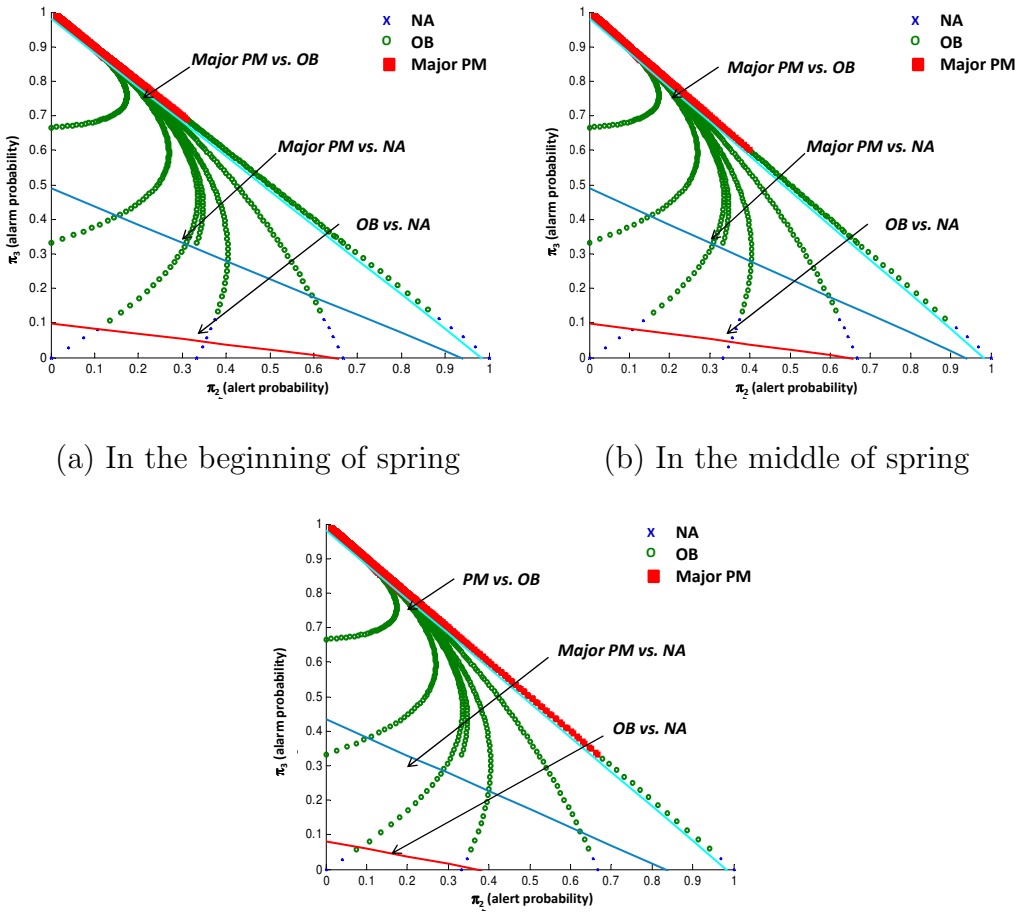
The decision rules will determine the best maintenance action that minimizes the operational costs using the current information of the system's physical condition and exogenous weather conditions. An advantage of this algorithm is that economic objectives and operational limits due to harsh weather conditions can be handled directly in a systematic manner. In addition, the rigorous model, which is based on optimization tasks, is always used and adapted online. Consequently, parameter tuning (or estimation) tasks can be considerably reduced.

VII.2. Preliminary results

To evaluate the performance of the real-time algorithm, we apply **Algorithm VII.1** to the same case study described in Section V.3. Suppose that the weather parameter estimates presented in Table 4 are perfectly accurate throughout the year, and thus the policy discussed in Section V.3 is the true optimal policy. In order to evaluate the performance of the real-time algorithm with the optimal policy as a benchmark, we superimpose the real-time decision boundaries on the optimal policy in Figs. 36 & 37. We omit the results for fall and winter because they are similar to the results for spring and summer. In the figures, the PM region from the real-time policy is defined as the area above the two lines of "PM vs. OB" and "PM vs. NA" (let us ignore the PM level for the time being; it is discussed in the next paragraph). Similarly, NA region is the area below the two lines of "PM vs. NA" and "OB vs. NA". The in-between area is OB region.

In most cases, the results from the real-time strategy are close to the optimal policy. We believe that the real-time strategy approximates the optimal policy well. We have a few remarks to explain the results of the real-time strategy.

- The decision boundaries to define the preference conditions over "PM vs. OB" and "PM vs. NA" approximate the optimal policy very well. On the contrary, there is a small discrepancy between the real-time policy and the optimal policy in separating NA area with OB area in the lower side of the figures. Recall that in the static CBM model, we derive necessary and sufficient conditions for the first two comparisons ("PM vs. OB" and "PM vs. NA"), but only the sufficient condition for NA to be preferred to OB is derived rigorously, and we approximate this sufficient condition as a necessary and sufficient condition in the real-time algorithm. One possible explanation for the discrepancy could be



(a) In the beginning of spring

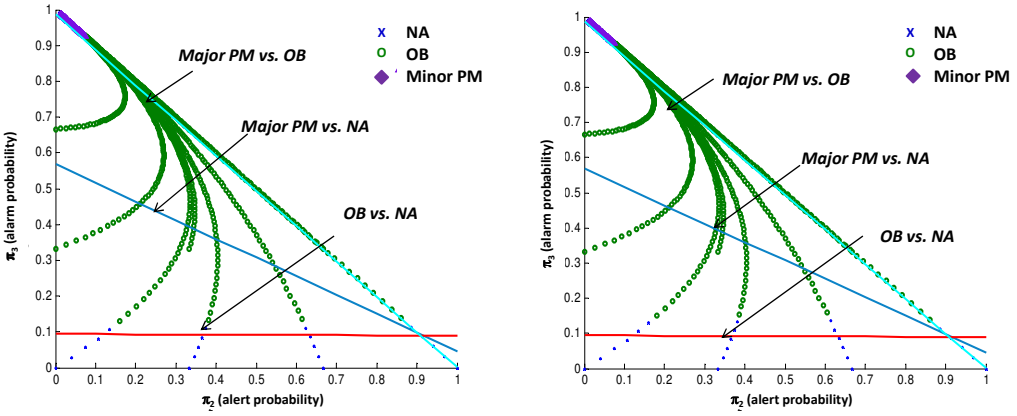
(b) In the middle of spring

(c) At the end of spring

Fig. 36. Approximate decision rules during spring season superimposed on optimal policy

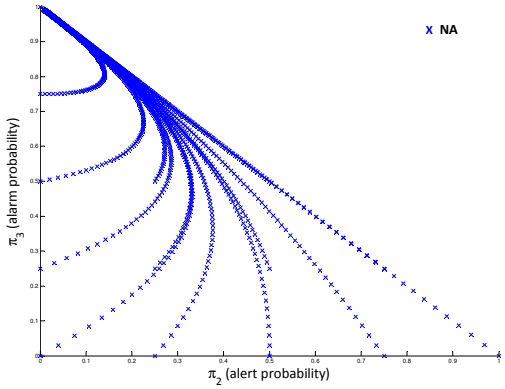
explained by this approximation.

- During the summer season (except the ending periods of the summer season), the optimal policy suggests taking the minor PM when the system condition is very bad, whereas the real-time algorithm suggests the major PM. In selecting the best PM level in the real-time algorithm, we use $C^R_{PM(m)}$ in (7.7). Note that the minor PM returns the system state to the *alert* state (e_2) in this example. But the bias at e_2, $b^R(e_2)$, turns out to be relatively higher, compared to the



(a) In the beginning of summer

(b) In the middle of summer



(c) At the end of summer

Fig. 37. Approximate decision rule during summer season superimposed on optimal policy

difference between the major PM cost and the minor PM cost. We conjecture that incorrect suggestions of the PM level from the real-time algorithm during summer periods may be explained by the imperfect parameter setting in cost factors and the elements of the transition matrix. We believe that much refined parameter setting will improve the quality of the real-time algorithm.

- At the end of summer storm seasons, weather conditions dramatically change in the next upcoming periods and better weather conditions are expected. Hence,

the real-time algorithm sets the best action as NA in all states as suggested by Step 5 of **Algorithm VII.1**, and leads to the same results as the dynamic optimal policy.

VII.3. Incorporation of optimization results in the simulation model

Recall that in our simulation platform, the experimental frame (EF) coupled model and operation and maintenance (OPMNT) coupled model are designed to be separated from the wind farm (WF) model. The advantage of this structure is that the addition of new O&M polices or changes in the current policies can be done independently without making changes to the other models.

The real-time O&M strategy can be implemented by embedding **Algorithm VII.1** in the OPMNT model in our simulation platform. At each decision point, the maintenance scheduler (MSCHEDR) in the OPMNT model will decide the proper maintenance action proposed by the real-time algorithm, based on the real-time weather information from the wind generator (WGENR) atomic model and the estimated state information from the state evaluation (STEVAL) atomic model. When PM or OB is recommended, the maintenance generator (MGENR) dispatches maintenance crew to the turbine. Once the repair is done, the report is sent to the transducer (TRANSD) atomic model to collect the performance information over the lifetime of wind turbines.

The EF model allows us to choose a simulation experiment of interest. Once the real-time algorithm is embedded in the OPMNT model, the real time O&M strategy can be selected in the EF. Finally, the collected information from the simulation runs can be compared with the results from other O&M strategies to validate the effectiveness of the real-time O&M strategy.

CHAPTER VIII

CONCLUSION

Based on a literature review and technical discussions with our industry partners, we observe that a comprehensive simulation tool and an effective O&M strategy are badly needed to address the approaching challenges faced by the global wind power industry. This chapter summarizes our proposed models and highlights the contributions of this dissertation. We conclude with suggestions for further research.

VIII.1. Summary

In this study we construct new probabilistic models for choosing cost-effective maintenance actions and scheduling adaptive on-site observations for wind turbine operations and maintenance. We develop dynamic optimal policies to respond to stochastic weather conditions. We also examine other aspects unique to turbine maintenance, including failure modes, partial as well as perfect repairs, and stochastic revenue losses.

We develop two optimization models by formulating the problem as a POMDP, which considers the costs associated with different actions and other critical aspects. To the best of our knowledge, the models proposed are the first mathematical models for condition-based wind turbine maintenance. In the first optimization model with homogeneous weather parameters, the static CBM model, we analytically derive a set of closed-form expressions for the optimal policy and show how the results can be utilized to solve large size problems. We extend the AM4R structure under weaker assumptions than previous literature, and also demonstrate the conditions under which this AM4R structure becomes an AM3R structure. We suggest that the

static CBM model is feasible for in-land wind farms where weather patterns are quite stable.

The second optimization model, the dynamic CBM model, uses heterogeneous parameters to reflect seasonal weather fluctuations commonly exhibited in very windy (usually high-altitude) or offshore sites. We show how to alter the decision rules based on weather characteristics to minimize O&M costs while maximizing the availability of wind power production. We introduce an algorithm to find the optimal policy of the dynamic CBM model using a backward dynamic programming. Applying the model to a case study we show that it can translate into considerable savings in operational costs and number of failures.

We also develop a simulation model which provides profound insights into the stochastic behavior of wind power systems. We use DEVS formalism because it provides a formal modeling and simulation framework based on dynamical systems theory and allows for hierarchical and modular model construction (Zeigler *et al.*, 2000). The simulation platform represents actual operations with sufficient details. We consider the condition of each component in a turbine and the correlation of wind generation among turbines, which is not generally addressed in the extant literature. We validate the simulation platform with several calibration criteria using field data and data from the literature. The criteria selected to account for both the dynamic response of turbines and their degradation include: capacity factor, availability of the wind power systems, and number of failures. We fine-tune a number of parameters so that the simulation results match the actual wind farm output.

The simulation model provides a tool for wind farm operators to select the most cost-effective O&M strategy. The two maintenance policies studied are SchM and CBM. Here, the SchM, which performs preventive repairs regularly, represents the standard practice in the industry, whereas the CBM is the strategy adopted

recently as condition monitoring techniques have advanced. In the CBM strategy, preventive repairs are carried out when sensors send alarm signals. The CBM strategy implemented in the simulation model is not the one based on the optimization models discussed in this study, but a simplified strategy. However, even with this simpler form of a CBM strategy, the implementation results demonstrate that the CBM strategy provides appreciable benefits over the SchM under all of the performance measures considered. For example, the failure frequency and the overall O&M costs drop by 11.7% and 31.5%, respectively, when the CBM is used instead of the SchM.

Finally, we present the integrated framework in which we incorporate the resulting optimization tools in our simulation model. We provide a real-time algorithm to decide a proper maintenance action based on the health status of a turbine component and the weather conditions. The set of decision rules consists of the specification of a pre-defined algorithm described by a set of if-then rules, which can be easily incorporated in the simulation model and easily implemented in practice. Determining the best maintenance policy will always rely on the available real-time weather forecasting data for the near future.

VIII.2. Suggestions for future research

We suggest extending the current framework to incorporate different levels of operations which include the electric grid and network, as shown in Fig. 38. At the finest granularity, we have already established modeling of each turbine (Fig. 38(c)). The finely grained models for wind turbines can be coupled to the model for wind farms (Fig. 38(b)) and also possibly to the models representing the grid and network (Fig. 38(a)). This extended framework will be able to evaluate the reliability of the grid with hybrid generation systems including conventional sources and storage.

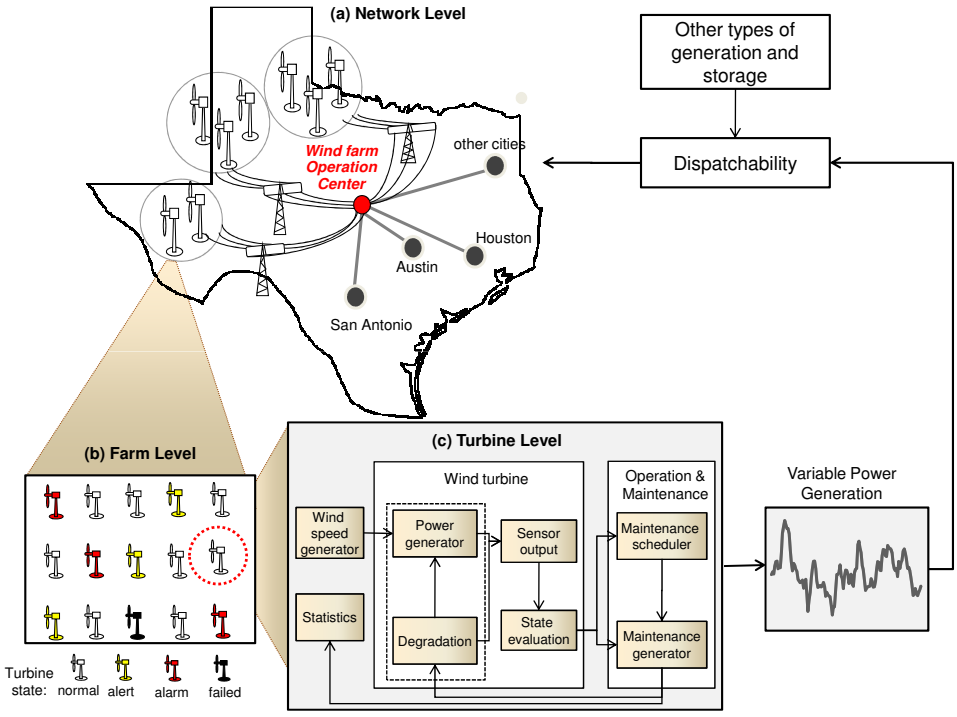


Fig. 38. Modeling and optimization of wind power systems at all different levels of operations: (a) power grid and network, (b) wind farm; (c) wind turbine

This vision opens up several research directions. First, the current simulation and optimization models can be extended to include degradation of multiple wind turbine components, such as gearboxes, generators and blades. The current study only considers gearbox maintenance because gearboxes are one of the most critical components and are prone to major failures. It would be interesting to study how robustly the recommended CBM policy performs when multiple components are considered.

Another direction is to develop a comprehensive methodology for multi-time scale decision-making. Note that the dispatching decisions at the network level (Fig. 38(c)) are analyzed on a short-term basis (hourly or daily), while the decisions related to

O&M (Fig. 38(b)) are made on a medium-term basis (weekly, bi-weekly, monthly, or seasonally). The two decision types affect one another, e.g., repairing a turbine may require temporarily stopping its production. No optimization tool is yet capable of handling the operations of a wind farm at different time scales. Comprehensive modeling and optimization efforts will be required to handle these inter-relationships.

The extended framework we have proposed provides a platform for a broad array of potential applications related to wind generation. For example, the simulation model can be extended to site evaluation (Acker *et al.*, 2007), correlation studies for multiple facilities (Wan *et al.*, 2003), wind power system reliability analysis (Karki and Patel, 2009, Wen *et al.*, 2009), and evaluation of generation adequacy of power systems (Karki and Billinton, 2004, Kaviani *et al.*, 2009). The framework can also be extended to assess generation capacity of hybrid models with different power generators such as conventional fuel-fired power, battery and wind energy (Nelson *et al.*, 2006).

In conclusion, acknowledging that the global economic recession is expected to continue, and that every dollar of private investment and government subsidies must "stretch", we have designed our proposed framework for easy understanding and application, for it, too, can be "stretched" to encompass a single farm, or multiple facilities.

REFERENCES

Acker, T.L., Williams, S.K., Duque, E.P., Brummels, G. and Buechler, J. (2007) Wind resource assessment in the state of Arizona: Inventory, capacity factor, and cost. *Renewable Energy*, **32**(9), 1453 – 1466.

American Wind Energy Association (2008) Http://awea.org/.

Andrawus, J.A., Watson, J. and Kishk, M. (2007) Modelling system failures to optimise wind farms. *Wind Engineering*, **31**, 503–522.

Billinton, R. and Li, Y. (2004) Incorporating multi-state unit models in composite system adequacy assessment, in *Proceedings of the eighth International Conference on Probabilistic Methods Applied to Power Systems*, Ames, IA, pp. 70–75.

Bowerman, B., O'Connell, R. and Koehler, A. (2005) *Forecasting, Time Series, and Regression*, Duxbury, Belmont, CA.

Bussel, G.V. (1999) The development of an expert system for the determination of availability and O&M costs for offshore wind farms, in *Proceedings of the 1999 European Wind Energy Conference and Exhibition*, Nice, France, pp. 402–405.

Caselitz, P. and Giebhardt, J. (2005) Rotor condition monitoring for improved operational safety of offshore wind energy converters. *Journal of Solar Energy Engineering*, **127**(2), 253–261.

Chattopadhyay, D. (2004) Life-cycle maintenance management of generating units in a competitive environment. *IEEE Transactions on Power Systems*, **19**, 1181 – 1189.

De Luna, X. and Genton, M.G. (2005) Predictive spatio-temporal models for spatially sparse environmental data. *Statistica Sinica*, **15**, 547–568.

Derman, C. (1963) On optimal replacement rules when changes of state are Markovian. Technical report, Mathematical Optimization Techniques R-396-PR, University of California Press, Berkeley, CA.

Di Fazio, A. and Russo, M. (2008) Wind farm modelling for reliability assessment. *IET Renewable Power Generation*, **2**, 239–248.

Ding, Y., Byon, E., Park, C., Tang, J., Lu, Y. and Wang, X. (2007) Dynamic data-driven fault diagnosis of wind turbine systems. *Lecture Notes in Computer Science*, **4487**, 1197–1204.

Echavarria, E., Hahn, B., van Bussel, G.J.W. and Tomiyama, T. (2008) Reliability of wind turbine technology through time. *Journal of Solar Energy Engineering*, **130**, 031005–1–031005–8.

Endrenyi, J., Aboresheid, S., Allan, R., Anders, G., Asgarpoor, S., Billinton, R., Chowdhury, N. and Dialynas, E. (2001) The present status of maintenance strategies and the impact of maintenance on reliability. *IEEE Transactions on Power Systems*, **16**(4), 638–646.

Endrenyi, J., Anders, G. and da Silva, A.L. (1998) Probabilistic evaluation of the effect of maintenance on reliability. An application to power systems. *IEEE Transactions on Power Systems*, **13**(2), 576–583.

Faulstich, S., Hahn, B., Jung, H., Rafik, K. and Ringhand, A. (2008) Appropriate failure statistics and reliability characteristics. Technical report, Fraunhofer Institute for Wind Energy, Bremerhaven, Germany.

Gebraeel, N. (2006) Sensory-updated residual life distributions for components with exponential degradation patterns. *IEEE Transactions on Automation Science and Engineering*, **3**, 382–393.

Ghasemi, A., Yacout, S. and Ouali, M.S. (2007) Optimal condition based maintenance with imperfect information and the proportional hazards model. *International Journal of Production Research*, **45**(4), 989–1012.

Gipe, P. (2000) *Wind Power, Revised Edition: Renewable Energy for Home, Farm, and Business*, Chelsea Green Publishing Company, White River Junction, VT.

Gneiting, T., Larson, K., Westrick, K., Genton, M. and Aldrich, E. (2007) Calibrated probabilistic forecasting at the stateline wind energy center: the regime-switching space-time method. *Journal of the American Statistical Association*, **101**, 968–979.

Haining, R. (2000) *Spatial Data Analysis in the Social and Environmental Sciences*, Cambridge University Press, Cambridge.

Hall, P. and Strutt, J. (2007) Probabilistic physics-of-failure models for component reliabilities using Monte Carlo simulation and Weibull analysis: a parametric study. *Reliability Engineering and System Safety*, **80**, 233–242.

Hendriks, H., Bulder, B., Heijdra, J., Pierik1, J., van Bussel, G., van Rooij, R., Zaaijer, M., Bierbooms, W., den Hoed, D., de Vilder, G., Goezinne, F., Lindo, M., van den Berg, R. and de Boer, J. (2000) DOWEC concept study; evaluation of wind turbine concepts for large scale offshore application, in *Proceedings of the Offshore Wind Energy in Mediterranean and other European Seas (OWEMES) Conference*, Siracusa, Italy, pp. 211–219.

Hoskins, R.P., Strbac, G. and Brint, A.T. (1999) Modelling the degradation of

condition indices. *IEE Proceedings - Generation, Transmission and Distribution*, **146**(4), 386–392.

Im, H., Rathouz, P. and Frederick, J. (2008) Space-time modeling of 20 years of daily air temperature in the Chicago metropolitan region. *Environmetrics*, **20**, 494–511.

Jirutitijaroen, P. and Singh, C. (2004) The effect of transformer maintenance parameters on reliability and cost: a probabilistic model. *Electric Power System Research*, **72**, 213–234.

Karki, R. and Billinton, R. (2004) Cost-effective wind energy utilization for reliable power supply. *IEEE Transactions on Energy Conversion*, **19**(2), 435–440.

Karki, R. and Patel, J. (2009) Reliability assessment of a wind power delivery system. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, **223**(1), 51–58.

Kaviani, A., Baghaee, H. and Riahy, G. (2009) Optimal sizing of a stand-alone wind/photovoltaic generation unit using particle swarm optimization. *Simulation*, **85**(2), 89–99.

Khan, M., Iqbal, M. and Khan, F. (2005) Reliability and condition monitoring of a wind turbine, in *Proceedings of the 2005 IEEE Canadian Conference on Electrical and Computer Engineering*, Saskatoon, Saskatchewan, pp. 1978–1981.

Kim, Y.H. and Thomas, L.C. (2006) Repair strategies in an uncertain environment: Markov decision process approach. *Journal of the Operational Research Society*, **57**, 957–964.

Leite, A., Borges, C. and Falcao, D. (2006) Probabilistic wind farms generation

model for reliability studies applied to Brazilian sites. *IEEE Transactions on Power Systems*, **21**, 1493 – 1501.

Lovejoy, W. (1987) Some monotonicity results for partially observed Markov decision processes. *Operations Research*, **35**, 736–742.

Magnano, L. and Boland, J. (2007) Generation of synthetic sequences of electricity demand: application in South Australia. *Energy*, **32**, 2230–2243.

Maillart, L.M. (2004) Optimal condition-monitoring schedules for multi-state deterioration systems with obvious failures. Technical report *778*, Department of Operations, Weatherhead School of Management, Cleveland, OH.

Maillart, L.M. (2006) Maintenance policies for systems with condition monitoring and obvious failures. *IIE Transactions*, **38**, 463–475.

Maillart, L.M. and Zheltova, L. (2007) Structured maintenance policies in interior sample paths. *Naval Research Logistics*, **54**, 645–655.

Mandl, P. (1959) Sur le comportement asmptotique des probabilities dans les ensembles des etats d'une chaine de Markov homgene (in Russian). *Casopis pro Pestovani Maematiky*, **84**(1), 140–149.

McMillan, D. and Ault, G.W. (2008) Condition monitoring benefit for onshore wind turbines: sensitivity to operational parameters. *IET Renewable Power Generation*, **2**(1), 60–72.

Miller, F.P., Vandome, A.F. and McBrewster, J. (2009) *Discrete Event Simulation*, Alphascript Publishing, Mauritius.

Negra, N., Holmstrøm, O., Bak-Jensen, B. and Sørensen, P. (2007) Wind farm generation assessment for reliability analysis of power systems. *Wind Engineering*, **31**(6), 383–400.

Nelson, D., Nehrir, M. and Wang, C. (2006) Unit sizing and cost analysis of stand-alone hybrid wind/PV/fuel cell power generation systems. *Renewable Energy*, **31**(10), 1641 – 1656.

NERC (2009) The 2009 long-term reliability assessment. Technical report, North American Electric Reliability Corporation (NERC), Washington, DC.

Nilsson, J. and Bertling, L. (2007) Maintenance management of wind power systems using condition monitoring systems-life cycle cost analysis for two case studies. *IEEE Transactions on Energy Conversion*, **22**(1), 223–229.

Norris, J. (1998) *Markov Chains*, Cambridge University Press, Cambridge.

Ohnishi, M., Kawai, H. and Mine, H. (1986) An optimal inspection and replacement policy under incomplete state information. *European Journal of Operational Research*, **27**, 117–128.

Pacot, C., Hasting, D. and Baker, N. (2003) Wind farm operation and maintenance management, in *Proceedings of the PowerGen Conference Asia*, Ho Chi Minh City, Vietnam, pp. 25–27.

Porta, J.M., Vlassis, N., Spaan, M.T. and Poupart, P. (2006) Point-based value iteration for continuous POMDPs. *The Journal of Machine Learning Research*, **24**, 2329 – 2367.

Puterman, M. (1994) *Markov Decision Process*, Wiley, New York.

Qian, S., Jiao, W., Hu, H. and Yan, G. (2007) Transformer power fault diagnosis system design based on the HMM method, in *Proceedings of the IEEE International Conference on Automation and Logistics*, Jinan, China, pp. 1077–1082.

Rademakers, L., Braam, H. and Verbruggen, T. (2003a) R&D needs for O&M of wind turbines. Technical report *ECN-RX–03-045*, ECN Wind Energy, Petten, The Netherlands.

Rademakers, L., Braam, H., Zaaijer, M. and van Bussel, G. (2003b) Assessment and optimisation of operation and maintenance of offshore wind turbines. Technical report *ECN-RX-03-044*, ECN Wind Energy, Petten, The Netherlands.

Ravindra, M. and Prakash, S. (2008) Generator system reliability analysis including wind generators using hourly mean wind speed. *Electric Power Components and Systems*, **36**, 1–16.

ReliaSoft BlocSim-7 software (2007) Http://www.reliasoft.com/products.htm/.

Ribrant, J. (2006) Reliability performance and maintenance - A survey of failures in wind power systems, Master's thesis, KTH School of Electrical Engineering, Stockholm.

Ribrant, J. and Bertling, L. (2007) Survey of failures in wind power systems with focus on Swedish wind power plants during 1997-2005. *IEEE Transactions on Energy conversion*, **22**, 167–173.

Rosenfield, D. (1976) Markovian deterioration with uncertain information. *Operations Research*, **24**, 141–155.

Ross, S. (1971) Quality control under Markovian deterioration. *Management Science*, **19**, 587–596.

Sayas, F.C. and Allan, R.N. (1996) Generation availability assessment of wind farms. *IEE Proceedings- Generation,. Transmission and Distribution*, **144**(5), 1253–1261.

Soares, L. and Medeiros, M. (2008) Modeling and forecasting short-term electricity load: a comparison of methods with an application with Brazilian data. *International Journal of Forecasting*, **24**, 630–644.

Sotiropoulos, F., P.Alefragis and Housos, E. (2007) A hidden Markov models tool for estimating the deterioration level of a power transformer, in *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation*, Patras, Greece, pp. 784–787.

Spaan, M.T.J. and Vlassis, N. (2005) Some monotonicity results for partially observed Markov decision processes. *Journal of Artificial Intelligence Research*, **24**, 195–220.

Thomas, L.C., Gaver, D.P. and Jacobs, P.A. (1991) Inspection models and their application. *IMA Journal of Mathematics Applied in Business & Industry*, **3**, 283–303.

Vachon, W. (2002) Long-term O&M Costs of wind turbines based on failure rates and repair costs, presented at the WINDPOWER 2002 Annual Conference.

Walford, C. (2006) Wind turbine reliability: understanding and minimizing wind turbine operation and maintenance costs. Technical report *SAND2006-1100*, Sandia National Laboratories, Albuquerque, NM.

Wan, Y., Milligan, M. and Parsons, B. (2003) Output power correlation between adjacent wind power plants. *Journal of Solar Energy Engineering*, **125**(4), 551–555.

Welte, T. (2009) Using state diagrams for modeling maintenance of deteriorating Systems. *IEEE Transactions on Power Systems*, **24**, 58 – 66.

Wen, J., Zheng, Y. and Donghan, F. (2009) A review on reliability assessment for wind power. *Renewable and Sustainable Energy Reviews*, **13**, 2485–2494.

West Texas Mesonet (2008) Http://www.mesonet.ttu.edu/.

Wiser, R. and Bolinger, M. (2008) Annual report on U.S. wind power installation, cost, performance trend: 2007. Technical report, U.S Department of Energy, Washington, DC.

Yang, F., Kwan, C. and Chang, C. (2008) Multiobjective evolutionary optimization of substation maintenance using decision-varying Markov model. *IEEE Transactions on Power Systems*, **23**, 1328 – 1335.

Zeigler, B., Kim, T. and Praehofer, H. (2000) *Theory of Modeling and Simulation*, Academic Press, Inc., Orlando, FL.

Zeigler, B. and Sarjoughian, H. (2003) Introduction to DEVS Modeling & Simulation with JAVA™: Developing Component-Based Simulation Models, http://www.acims.arizona.edu.

Zhang, X., Zhang, J. and Gockenbach, E. (2009) Reliability centered asset management for medium-voltage deteriorating electrical equipment based on Germany failure statistics. *IEEE Transactions on Power Systems*, **24**, 721 – 728.

Zhou, W., Yang, H. and Fang, Z. (2006) Wind power potential and characteristic analysis of the Pearl River Delta region, China. *Renewable and Sustainable Energy Reviews*, **31**(6), 739–753.

APPENDIX A

MATHEMATICAL EXPRESSIONS OF THE ATOMIC MODELS IN PARALLEL DEVS

A.1. Power Generator (PWRGEN) Model

In this section a mathematical definition of the PWRGEN atomic model is provided. A "cut_off" boolean variable is used to notify when the wind speed is within a specified threshold (true) or not (false). Another boolean variable called "degradation" is used to notify when degradation has occurred (true) in the component or not (false). STI is used to denote a short time interval. An entity called msg is used to carry out the output information of the model.

$$DEVS_{PWRGEN} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.1}$$

where,

$IPorts = \{$"turb_on_off", "wind_in", "deg_in", "corr_mnt", "prev_mnt", "obsv", "req_status"$\}$, where $X_{turb\_on\_off} = V_1$, $X_{wind\_in} = V_2$, $X_{deg\_in} = V_3$, $X_{corr\_mnt} = V_4$, $X_{prev\_mnt} = V_5$, $X_{obsv} = V_6$, $X_{req\_status} = V_7$ are arbitrary sets;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$"pwr_out", "deg_on_off", "status_out"$\}$, where Y_{pwr_out}, $Y_{deg\_on\_off}$, and Y_{status_out} are arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"off_normal", "off_normal_waiting", "on_normal","off_alert", "off_alert_waiting", "on_alert", "off_alarm", "off_alarm_waiting", "on_alarm",

"failed", "report_status_n", "report_status_t", "report_status_m" $\} \times \Re_0^+ \times V_1 \times V_2 \times V_3 \times V_4 \times V_5 \times V_6 \times V_7$ is the set of sequential states.

***External Transition Function*:**

$$\delta_{ext}((phase, \sigma), e, (p, v))$$

$$= (\text{"off\_normal"}, \infty), \quad \text{if} \begin{cases} p = \text{"turb\_on\_off"}; \\ phase = \text{"off\_alert"} \wedge p = \text{"prev\_mnt"}; \\ phase = \text{"off\_alarm"} \wedge p = \text{"prev\_mnt"}; \\ phase = \text{"failed"} \wedge p = \text{"corr\_mnt"}; \\ phase = \text{"on\_normal"} \wedge p = \text{"turb\_on\_off"}; \\ phase = \text{"on\_normal"} \wedge p = \text{"prev\_mnt"}. \end{cases}$$

$$= (\text{"off\_normal\_waiting"}, \infty), \quad \text{if} \begin{cases} phase = \text{"off\_normal"} \wedge \\ \qquad \text{p} = \text{"turb\_on\_off"} \wedge \\ \qquad \text{cut\_off} = \text{false}; \\ phase = \text{"on\_normal"} \wedge \\ \qquad \text{p} = \text{"wind\_in"} \wedge \\ \qquad \text{cut\_off} = \text{false}. \end{cases}$$

$$= (\text{"on\_normal"}, \infty), \quad \text{if} \begin{cases} phase = \text{"off\_normal"} \wedge p = \text{"turb\_on\_off"} \\ \qquad \wedge\ cut\_off = true; \\ phase = \text{"off\_normal\_waiting"} \wedge \\ \qquad \text{p} = \text{"wind\_in"} \wedge \\ \qquad \text{cut\_off} = \text{true}. \end{cases}$$

$$= (\text{"off\_alert"}, \infty), \quad \text{if} \begin{cases} phase = \text{"on\_alert"} \wedge p = \text{"turb\_on\_off"}; \\ phase = \text{"on\_alert"} \wedge p = \text{"prev\_mnt"}. \end{cases}$$

$$= (\text{“off\_alert\_waiting”}, \infty), \quad \text{if} \begin{cases} phase = \text{“off\_alert”} \wedge p = \text{“turb\_on\_off”} \\ \qquad \wedge\, cut\_off = false; \\ phase = \text{“on\_alert”} \wedge p = \text{“wind\_in”} \\ \qquad \wedge\, cut\_off = false. \end{cases}$$

$$= (\text{“on\_alert”}, \infty), \quad \text{if} \begin{cases} phase = \text{“off\_alert”} \wedge p = \text{“turb\_on\_off”} \\ \qquad \wedge\, cut\_off = true; \\ phase = \text{“off\_alert\_waiting”} \wedge p = \text{“wind\_in”} \\ \qquad \wedge\, cut\_off = true; \\ phase = \text{“on\_normal”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true. \end{cases}$$

$$= (\text{“off\_alarm”}, \infty), \quad \text{if} \begin{cases} phase = \text{“on\_alarm”} \wedge p = \text{“turb\_on\_off”}; \\ phase = \text{“on\_alarm”} \wedge p = \text{“prev\_mnt”}. \end{cases}$$

$$= (\text{“off\_alarm\_waiting”}, \infty), \quad \text{if} \begin{cases} phase = \text{“off\_alarm”} \wedge p = \text{“turb\_on\_off”} \\ \qquad \wedge\, cut\_off = false; \\ phase = \text{“on\_alarm”} \wedge p = \text{“wind\_in”} \\ \qquad \wedge\, cut\_off = false. \end{cases}$$

$$= (\text{“on\_alarm”}, \infty), \quad \text{if} \begin{cases} phase = \text{“off\_alarm”} \wedge p = \text{“turb\_on\_off”} \\ \qquad \wedge\, cut\_off = true; \\ phase = \text{“off\_alarm\_waiting”} \wedge p = \text{“wind\_in”} \\ \qquad \wedge\, cut\_off = true; \\ phase = \text{“on\_alert”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true; \\ phase = \text{“on\_normal”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true. \end{cases}$$

$$= (\text{“failed”}, \infty), \quad \text{if} \begin{cases} phase = \text{“on\_normal”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true; \\ phase = \text{“on\_alert”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true; \\ phase = \text{“on\_alarm”} \wedge p = \text{“deg\_in”} \\ \qquad \wedge\, degradation = true. \end{cases}$$

$$= (\text{“report\_status\_n”}, STI), \quad \text{if} \begin{cases} phase = \text{“on\_normal”} \wedge \\ p = \text{“req\_status”}; \end{cases}$$

$$= (\text{“report\_status\_t”}, STI), \quad \text{if} \begin{cases} phase = \text{“on\_alert”} \wedge \\ p = \text{“req\_status”}; \end{cases}$$

$$= (\text{“report\_status\_m”}, STI), \quad \text{if} \begin{cases} phase = \text{“on\_alarm”} \wedge \\ p = \text{“req\_status”}. \end{cases}$$

$$= (phase, \sigma - e), \quad \text{otherwise.}$$

***Internal Transition Function*:**

$\delta_{int}(phase, \sigma)$

$= (\text{“on\_normal”}, STI\ ), \quad \text{if } phase = \text{“report\_status\_n”}$

$= (\text{“on\_alert”}, STI\ ), \quad \text{if } phase = \text{“report\_status\_t”}$

$= (\text{“on\_alarm”}, STI\ ), \quad \text{if } phase = \text{“report\_status\_m”}$

***Confluence Function*:**

$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x).$

***Output Function*:**

$\lambda(phase, \sigma)$

$$= (\text{“pwr\_out”}, msg) \quad \text{if} \begin{cases} phase = \text{“off\_normal”}; \\ phase = \text{“off\_alert”}; \\ phase = \text{“off\_alarm”}; \\ phase = \text{“failed”}. \end{cases}$$

$$= (\text{“deg\_on\_off”}, msg) \quad \text{if} \begin{cases} phase = \text{“off\_normal”}; \\ phase = \text{“off\_normal\_waiting”}; \\ phase = \text{“off\_alert”}; \\ phase = \text{“off\_alert\_waiting”}; \\ phase = \text{“off\_alarm”}; \\ phase = \text{“off\_alarm\_waiting”}; \\ phase = \text{“failed”}. \end{cases}$$

$$= (\text{“status\_out”}, msg) \quad \text{if} \begin{cases} phase = \text{“report\_status\_n”}; \\ phase = \text{“report\_status\_t”}; \\ phase = \text{“report\_status\_m”}. \end{cases}$$

***Time Advance Function*:**

$$ta(phase, \sigma) = \sigma$$

A.2. Component Degradation (CMPDEG) Atomic Model

Similarly the CMPDEG atomic model is defined in Parallel DEVS. The boolean variable deg is used to denote a change in degradation (true) or no change (false). Another boolean is defined named LTI which assumes the value of true when a large time interval is elapsed.

$$DEVS_{CMPDEG} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.2}$$

where,

$IPorts = \{$"wind_on_off", "maint_on_off", "manual_on_off"$\}$, where

$X_{wind\_on\_off} = V_1$, $X_{maint\_on\_off} = V_2$, and $X_{manual\_on\_off} = V_3$ are arbitrary sets;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$"deg_out", "status_out"$\}$, where Y_{deg_out}, and Y_{status_out} are arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"passive", "active", "passive_wind", "report_status", "report_deg", "passive_service"$\} \times \Re_0^+ \times V_1 \times V_2 \times V_3$ is the set of sequential states.

External Transition Function:

$$\delta_{ext}((phase, \sigma, status, deg), e, (p, v))$$

$$= (\text{“passive”}, \infty, \text{status}, \text{deg}), \quad \text{if} \begin{cases} p = \text{“manual\_on\_off”}; \\ phase = \text{“active”} \wedge \\ \qquad p = \text{“manual\_on\_off”}; \\ phase = \text{“passive\_service”} \wedge \\ \qquad p = \text{“maint\_on\_off”}. \end{cases}$$

$$= (\text{“active”}, \infty, \text{status}, \text{deg}), \quad \text{if} \begin{cases} phase = \text{“passive”} \wedge \\ \qquad p = \text{“manual\_on\_off”}; \\ phase = \text{“passive\_wind”} \wedge \\ \qquad p = \text{“wind\_on\_off”}. \end{cases}$$

$$= (\text{“passive\_wind”}, \infty, \text{status}, \text{deg}), \quad \text{if } phase = \text{“active”} \wedge$$

$p = $ “wind_on_off”

$$= (\text{“passive\_service”}, \infty, \text{status, deg}), \quad \text{if} \begin{cases} phase = \text{“active”} \wedge \\ \qquad p = \text{“maint\_on\_off”}; \\ phase = \text{“passive\_wind”} \wedge \\ \qquad p = \text{“maint\_on\_off”}. \end{cases}$$

$$= (phase, \sigma - e), \quad \text{otherwise.}$$

***Internal Transition Function*:**

$$\delta_{int}((phase, \sigma, status, deg), e, (p, v))$$

$$= (\text{“report\_status”}, STI, \text{status, deg}), \quad \text{if} \begin{cases} phase = \text{“active”} \wedge \\ \qquad LTI = true; \\ phase = \text{“passive\_wind”} \wedge \\ \qquad LTI = true. \end{cases}$$

$$= (\text{“report\_deg”}, STI, \text{status, deg}), \quad \text{if} \begin{cases} phase = \text{“active”} \wedge deg = true; \\ phase = \text{“passive\_wind”} \wedge \\ \qquad deg = true. \end{cases}$$

$$= (\text{“active”}, \infty, \text{status, deg}), \quad \text{if} \begin{cases} phase = \text{“report\_status”} \wedge \\ \qquad status = true; \\ phase = \text{“report\_deg”} \wedge \\ \qquad status = true. \end{cases}$$

$$= (\text{“passive\_wind”}, \infty, \text{status, deg}), \quad \text{if} \begin{cases} phase = \text{“report\_status”} \wedge \\ \qquad status = false; \\ phase = \text{“report\_deg”} \wedge \\ \qquad status = false. \end{cases}$$

***Confluence Function*:**

$$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x).$$

***Output Function*:**

$$\lambda(phase, \sigma, status, deg)$$

$$= (\text{“status\_out”}, msg) \quad \text{if } phase = \text{“report\_status”}$$

$$= (\text{“deg\_out”}, msg) \quad \text{if } phase = \text{“report\_deg”}$$

***Time Advance Function*:**

$$ta(phase, \sigma, status, deg) = \sigma$$

A.3. Sensor (SENSR) Atomic Model

We consider a SENSR with the input and outports ports shown in Fig. 39. The model has two input ports, "sensor_on_off" and "status_in". When an input is received at the "sensor_on_off" input port the model transitions from the *off* state to the *on* state. A change from the *on* state to the *retrieving_info* state will occur if an input is received at the "status_in" input port. After retrieving the information the model goes back to the *on* state. The SENSR has one output port, named "status_out". This outport is used to send information to the STEVAL model.

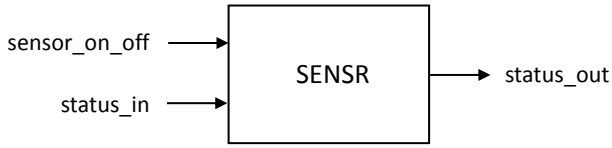


Fig. 39. Sensor with input and output ports

The Sensor (SENSR) atomic model has 3 basic states shown in Fig. 40; *off*, *on*, and *retrieving_info*.

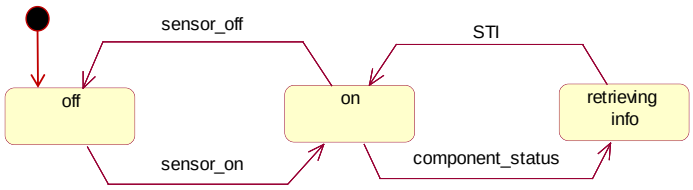


Fig. 40. Sensor state transition diagram

Mathematically, an atomic model for *Component_sensor* in Parallel DEVS can be defined as follows:

$$DEVS_{SENSR} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.3}$$

where,

$IPorts = \{$"sensor_on_off", "status_in"$\}$, where $X_{sensor\_on\_off} = V_1$ and $X_{status\_in} = V_2$ are arbitrary sets;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$"status_out"$\}$, where Y_{status_out} is an arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"off", "on", "retrieving_info"$\} \times \Re_0^+ \times V_1 \times V_2$ is the set of sequential states.

External Transition Function:

$\delta_{ext}((phase, \sigma, info), e, (p, v))$

$=$ ("off", ∞, info), if $p =$ "sensor_on_off"

$=$ ("off", ∞, info), if $phase =$ "on" $\wedge$ $p =$ "sensor_on_off"

$=$ ("on", ∞, info), if $phase =$ "off" $\wedge$ $p =$ "sensor_on_off"

$=$ ("retrieving_info", STI, info), if $phase =$ "on" $\wedge$ $p =$ "status_in"

$turbineStatus =$ getCurrentStatus(info);

$= (phase, \sigma - e)$, otherwise.

Internal Transition Function:

$\delta_{int}((phase, \sigma, info), e, (p, v))$

$=$ ("on", ∞, $status$), if $phase =$ "retrieving_info" $\wedge$ $STI = true$

Confluence Function:

$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x)$.

Output Function:

$\lambda(phase, \sigma, info)$

$=$ ("status_out", msg) if $phase =$ "retrieving_info"

Time Advance Function:

$ta(phase, \sigma) = \sigma$

A.4. Smart Sensor (SMSENSR) Atomic Model

The Smart Sensor (SMSENSR) atomic model has 4 basic states; *off*, *on*, *check_status* and *retrieving_info*. We consider a SMSENSR with the input and outports ports shown in Fig. 41. The model has three input ports, "sensor_on_off", "turbine_status", and "status_in".

Fig. 41. Smart sensor with input and output ports

When an input is received at the "sensor_on_off" input port the model transitions from the *off* state to the *on* state. A change from the *on* state to the *check_status* state will occur if a message is received at the "turbine_status" input port. This message is a request for additional information about the component status. The SMSENSR atomic model is in charge of obtaining the information needed by sending a message to the PWRGEN atomic model. A transition from the *check_status* state to the "retrieving_info" state will occur when a message is received at the "status_in"

input port containing the information requested. After retrieving the information the model goes back to the *on* state.

The SMSENSR has two output ports, named "status_out" and "request_status". The "status_out" output port is used to send requested status information to the MSCHEDR atomic model and the "request_status" output port is used to request the real status of the PWRGEN atomic model. The operation of the SMSENSR model is depicted in Fig. 42.

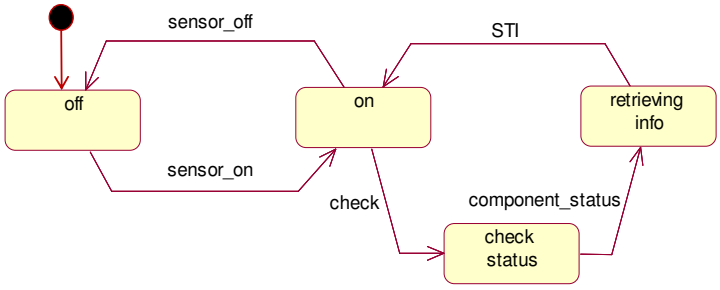


Fig. 42. Smart sensor state transition diagram

Mathematically, an atomic model for *Smart_sensor* in Parallel DEVS can be defined as follows:

$$DEVS_{SMSENSR} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.4}$$

where,

$IPorts = \{$"sensor_on_off", "status_in", "turbine_status"$\}$, where $X_{sensor\_on\_off} = V_1$, $X_{status\_in} = V_2$ and $X_{turbine\_status} = V_3$ are arbitrary sets;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$"status_out", "check_status"$\}$, where Y_{status_out} and Y_{check_status} are arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"off", "on", "check_status", "retrieving_info"$\} \times \Re_0^+ \times V_1 \times V_2 \times V_3$ is the set of sequential states.

External Transition Function:

$\delta_{ext}((phase, \sigma, info), e, (p, v))$

$= ($"off", $\infty, info)$, if $p =$ "sensor_on_off"

$= ($"off", $\infty, info)$, if $phase =$ "on" $\wedge$ $p =$ "sensor_on_off"

$= ($"on", $\infty, info)$, if $phase =$ "off" $\wedge$ $p =$ "sensor_on_off"

$= ($"check_status", $\infty, info)$, if $phase =$ "on" $\wedge$ $p =$ "turbine_status"

$= ($"retrieving_info", STI, $info)$, if $phase =$ "check_status" $\wedge$ $p =$ "status_in"

$turbineStatus =$ getCurrentStatus(info);

$= (phase, \sigma - e)$, otherwise.

Internal Transition Function:

$\delta_{int}((phase, \sigma, info), e, (p, v))$

$= ($"on", $\infty, info)$, if $phase =$ "retrieving_info" $\wedge$ $STI = true$

Confluence Function:

$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x)$.

Output Function:

$\lambda(phase, \sigma, info)$

$= (\text{“request\_status”}, request)$ if $phase =$ “check_status”

$= (\text{“status\_out”}, msg)$ if $phase =$ “retrieving_info”

Time Advance Function:

$ta(phase, \sigma) = \sigma$

A.5. State Evaluation (STEVAL) Atomic Model

The State evaluation (STEVAL) atomic model has 3 basic states; *off*, *on*, and *retrieving_info*. We consider a STEVAL with the input and outports ports shown in Fig. 43. The model has two types of input ports, namely; “se_on_off” and “sensor_x_in”. The number of input ports of type “sensor_x_in” depends on the number of sensors existing in the component.

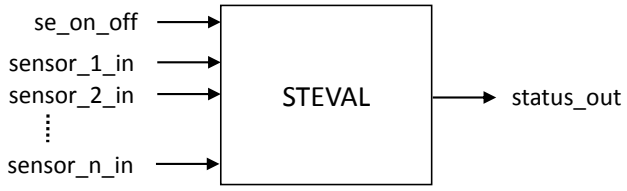


Fig. 43. State evaluation with input and output ports

When an input is received at the “se_on_off” input port the model transitions from the *off* state to the *on* state. A change from the *on* state to the *retrieving_info* state will occur if an input is received in one of the “sensor_x_in” input ports. After retrieving the information the model goes back to the *on* state.

The STEVAL has one output port, named “status_out”. This outport is used to send information to the MSCHEDR model. The operation of the STEVAL model is

depicted in Fig. 44.

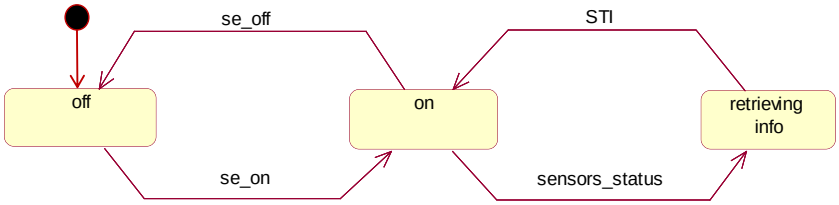


Fig. 44. State evaluation state transition diagram

Mathematically, an atomic model for *State_evaluation* in Parallel DEVS can be defined as follows:

$$DEVS_{STEVAL} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.5}$$

where,

$IPorts = \{$"se_on_off", "sensor_1_in", "sensor_2_in", ... , "sensor_n_in"$\}$, where $X_{se\_on\_off} = V_1$, $X_{sensor\_1\_in} = V_2$, $X_{sensor\_2\_in} = V_3$, ... , $X_{sensor\_n\_in} = V_{n+1}$ are arbitrary sets;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$"status_out"$\}$, where Y_{status_out} is an arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"off", "on", "retrieving_info"$\} \times \Re_0^+ \times V_1 \times V_2 \times$ .... $\times V_{n+1}$ is the set of sequential states.

External Transition Function:

$\delta_{ext}((phase, \sigma, info), e, (p, v))$

$= (\text{“off”}, \infty, info), \quad \text{if } p = \text{“se\_on\_off”}$

$= (\text{“off”}, \infty, info), \quad \text{if } phase = \text{“on”} \wedge p = \text{“se\_on\_off”}$

$= (\text{“on”}, \infty, info), \quad \text{if } phase = \text{“off”} \wedge p = \text{“se\_on\_off”}$

$= (\text{“retrieving\_info”}, STI, info), \quad \text{if } phase = \text{“on”} \wedge p = \text{“sensor\_}x\text{\_in”}$

$turbineStatus = \text{getSensorStatus(info);}$

$= (phase, \sigma - e, info)$, otherwise.

Internal Transition Function:

$\delta_{int}((phase, \sigma, info), e, (p, v))$

$= (\text{“on”}, \infty, info), \quad \text{if } phase = \text{“retrieving\_info”} \wedge STI = true$

Confluence Function:

$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x).$

Output Function:

$\lambda(phase, \sigma, info)$

$= (\text{“status\_out”}, msg) \quad \text{if } phase = \text{“retrieving\_info”}$

Time Advance Function:

$ta(phase, \sigma, info) = \sigma$

A.6. Maintenance Scheduler (MSCHEDR) Atomic Model

The MSCHEDR atomic model is in charge of determining maintenance schedules for the system components. The modeler can implement any scheduling algorithm

using this model. The MSCHEDR atomic model has 3 basic states; "passive", "update_schedule", and "active". The model has only one input port, called "status_in". Fig. 45 shows the input and output ports for the model.

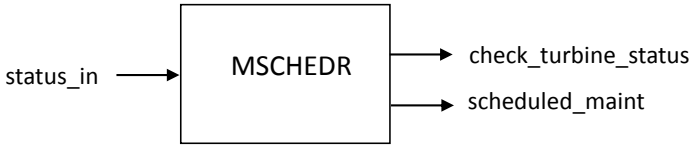


Fig. 45. Maintenance scheduler with input and output ports

A transition to the "active" state occurs when the model is on an "passive" state and a message is received at the "status_in" input port. A method, named getMaintSchedule(); takes the information provided and perform the scheduling using the algorithm defined by the modeler. If the scheduling is successfully performed, the MSCHEDR atomic model transitions to the "update_schedule" state where the schedules of the resources seized to perform maintenance are updated. After completing the schedules updates the MSCHEDR atomic model transitions to the "idle" state. The model will also transition back to the "idle" if it is on the "active" state and no schedule for maintenance is assigned or if additional information is needed to make the decision.

The MSCHEDR has two types of output ports, namely; "check_turbine_status" and "scheduled_maint". Messages are sent using the "check_turbine_status" output port when additional information from one components is needed to schedule maintenance. The "scheduled_maint" output port is used to send information to the Maintenance generator (MGENR) atomic model. The operation of the MSCHEDR atomic model is depicted in Fig. 46.

Mathematically, the *Maintenance_scheduler* model can be represented as follows.

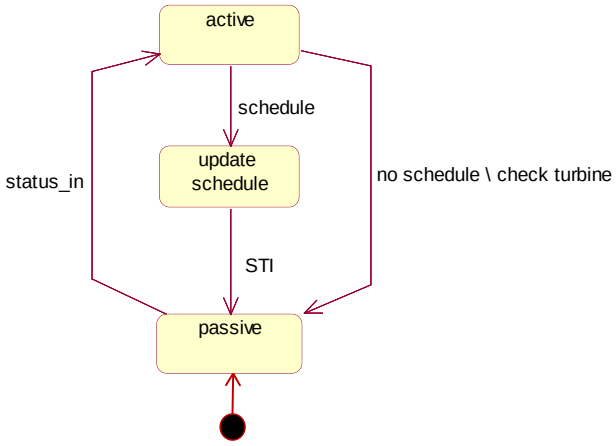


Fig. 46. Maintenance scheduler state transition diagram

$$DEVS_{MSCHEDR} = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{A.6}$$

where,

$IPorts = \{$"status_in"$\}$, where $X_{status\_in} = V_1$ is an arbitrary set;

$X_M = \{(p, v) | p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$OPorts = \{$ "check_turbine_status", "scheduled_maint"$\}$, where $Y_{check\_turbine\_status}$, $Y_{scheduled\_maint}$ are arbitrary sets;

$Y_M = \{(p, v) | p \in OPorts, v \in Y_p\}$ is the set of output ports and values; and

$S = \{$"active", "update_schedule", "passive"$\} \times \Re_0^+ \times V_1$ is the set of sequential states.

External Transition Function:

$\delta_{ext}((phase, \sigma, info), e, (p, v))$

$= (\text{“active”}, STI, info), \quad \text{if } phase = \text{“passive”} \wedge p = \text{“status\_in”}$

$maintenance = \text{scheduleMaintenance}(info);$

$= (phase, \sigma - e, call_i), \text{ otherwise.}$

Internal Transition Function:

$\delta_{int}((phase, \sigma, info), e, (p, v))$

$= (\text{“update\_schedule”}, STI, info), \quad \text{if } phase = \text{“active”} \wedge schedule = true$

$$= (\text{“passive”}, \infty, info), \quad \text{if} \begin{cases} phase = \text{“update\_schedule”} \wedge STI = true; \\ phase = \text{“active”} \wedge schedule = false; \\ phase = \text{“active”} \wedge check\_turbine = true. \end{cases}$$

Confluence Function:

$\delta_{con}(s, ta(s), x) = \delta_{ext}(\delta_{int}(s), 0, x).$

Output Function:

$\lambda(phase, \sigma, info)$

$= (\text{check\_turbine\_status}, msg) \quad \text{if } phase = \text{“active”} \wedge check\_turbine = true,$

$= (\text{sched\_maint}, maintenance) \quad \text{if } phase = \text{“update\_schedule”}$

Time Advance Function:

$ta(phase, \sigma, call_i) = \sigma$

VITA

Eunshin Byon received her B.S. in 1994, and M.S. (Honors) in 1996, in Industrial and Systems Engineering from Korea Advanced Institute of Science and Technology (KAIST), South Korea. She joined Texas A&M University in Fall 2005 for graduate studies in industrial engineering, where she was associated with the Advanced Metrology Lab. Eunshin's research interests include operations and management of wind power systems, statistical modeling and analysis for complex systems, discrete event simulation, and simulation-based optimization. She will be joining the Department of Industrial and Systems Engineering of the Texas A&M University as a research associate in June 2010.

Eunshin may be reached at her work address:

Department of Industrial and Systems Engineering

241 Zachry Engineering Research Center

Texas A&M University, 3131 TAMU

College Station, TX. 77843-3131

This document was typeset in LaTeX by the author.