# PERFORMANCE-CENTRED MAINTENANCE PROBLEMS:

# MODELLING and HEURISTICS

by

**Junchi Tan**

Department of Management Science

University of Strathclyde

A thesis presented in fulfilment of the requirement for the degree of

*Doctor of Philosophy*

*(Management Science)*

2019

# Declarations of Authenticity and Author Rights

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:

Date:

# Abstract

This study focuses on aligning the business functions of maintainers and operators in the manufacturing industry for better costs/profits. In practice, maintainers and operators undertake maintenance operations management and production operations management respectively; such two types of operations management are closely intertwined with each other and hence should not be considered independently. In the maintenance planning literature, however, the balance between machine utilisation and increased risk of failure is rarely thoroughly discussed. Motivated by the maintenance planning problem in a large scale British coal-fired power plant, this study aims at developing a theoretical framework which facilitates aligning the business functions of maintainers and operators for better costs/profits in a relatively generic manufacturing industrial setting.

Initially we consider a generic multi-asset production system. In such system we investigate the fundamental trade-offs between the decision making of maintainers and operators, and we further analyse how such trade-offs are underpinned by the existence of contracted period for sales and the associated potentially high penalty cost; based on such problem structure elicitation, we then develop a maintenance approach which integrates the operators' decision making as part of the maintainers' decision making in a conceptual framework and then further mathematically formulates such integrated maintenance planning problem as a Markov decision process (MDP). Such maintenance approach not only facilitates maintenance planning optimisation given existing machine utilisation behaviours, but also facilitates machine utilisation behaviour improvement as researchers/practitioners can conduct what-if analysis by changing the integrated utilisation behaviours in the MDP model.

Next we consider a multi-level hierarchical physical structure which involves multiple aforementioned multi-asset production systems; such hierarchical structure is shared by many industrial cases. We scale up the MDP model to capture such complex hierarchical structure, in the context of the coal-fired power plant case. The resulted mathematical problem is too complex to be solved by exact methods, and we therefore develop a set of heuristics to solve the problem: we select a simulation-based computation heuristic and a value function approximation method from literature, and we further combine them with our own designed decomposition method and parameters number bounding method. We further discuss how the scaled-up mathematical model and heuristics can be applied to other industrial cases of interest. Finally, we conduct numerical tests to demonstrate the practical value of our maintenance approach, mathematical model and heuristics.

# Acknowledgements

I would like to present my most sincere thanks to my supervisors Prof. Tim Bedford and Dr. Kerem Akartunali for their valuable guidance, support and sharing of wisdom. I would also like to thank Dr. Euan Barlow for helpful discussions and assistance. I would also like to extend my gratitude to the Management Science Department Office and my colleagues who offered me kind help and support.

I would like to thank my family for their encouragement and strong support throughout the years.

Junchi Tan

*Glasgow, 2018*

# Abbreviations

| | |
|---|---|
| MDPs | Markov decision processes |
| DP | Dynamic programming |
| VI | Value iteration |
| RL | Reinforcement learning |

# Nomenclature

### Notations first introduced in Chapter 2

| | |
|---|---|
| $T$ | Time length of the maintenance planning period |
| $T_C$ | Time length of the contracted period |
| $t$ | Time-step |
| $\mathbf{x}$ | System state |
| $D$ | Demand |
| $wr$ | Work-rate of system |
| $\mathbf{c}$ | Condition |
| $\mathbf{p}$ | Performance |
| $n$ | Index of asset |
| $N$ | Total amount of assets in the system |
| $S$ | State space of MDP |
| $a$ | Action |
| $A$ | Action space of MDP |
| $R(\bullet)$ | Reward function of MDP |

| | |
|---|---|
| $P(\bullet)$ | Transition probability of MDP |
| $\pi$ | Policy of MDP |
| $V$ | Expected total discount rewards |
| $R_C(\bullet)$ | Reward function in the contracted period |
| $R_{NC}(\bullet)$ | Reward function in the non-contracted period |

**Notations first introduced in Chapter 3**

| | |
|---|---|
| $V^*$ | Optimal value of MDP |
| $Q$ | An estimation of $V$ value |

**Notations first introduced in Chapter 4**

| | |
|---|---|
| $K$ | Total number of units in the plant |
| $OH$ | Time-length an overhaul takes |
| $\mathbf{x}_M$ | State of the generic mill |
| $\mathbf{x}_P$ | State of the plant |
| $\mathbf{x}_{k,n}$ | State of the mill indexed as mill $n$ in the unit indexed as unit $k$ in the power plant |
| $T_S$ | Time-length of the short-term part of the non-contracted period |
| $T_L$ | Time-length of the long-term part of the non-contracted period |
| $h$ | Time-step in Stage (3) of the three-stage hybrid MDP model |
| $ts$ | Time-step in Stage (2) of the three-stage hybrid MDP model |
| $oh$ | Total number of time-steps an undergoing overhaul has lasted |

| | |
|---|---|
| $tc$ | Time-step in Stage (1) of the three-stage hybrid MDP model |
| $\mathbf{c}_{k,n}$ | Condition of mill $n$ in unit $k$ |
| $\mathbf{p}_{k,n}$ | Performance of mill $n$ in unit $k$ |
| $\overline{Q}$ | Output value of a chosen quadratic function |
| $w$ | Free parameter in a chosen quadratic function |

# Contents

**Declarations of Authenticity and Author Rights** **i**

**Abstract** **ii**

**Acknowledgements** **iv**

**Abbreviations** **v**

**Nomenclature** **vi**

**Contents** **ix**

**1 Introduction** **1**

1.1 An overview of machine maintenance approaches . . . . . . . . . . . . . . . . 4

1.1.1 CM approaches . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

1.1.2 PM approaches . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

1.1.2.1 TBM approaches . . . . . . . . . . . . . . . . . . . . . . . 8

1.1.2.2 CBM approaches . . . . . . . . . . . . . . . . . . . . . . . 12

1.2 Further literature review on relevant studies . . . . . . . . . . . . . . . . . . 19

1.2.1 CBM studies involving operational performance . . . . . . . . . . . 20

1.2.2 Maintenance-production joint scheduling . . . . . . . . . . . . . . . 22

1.3 Thesis structure and research objectives . . . . . . . . . . . . . . . . . . . . 23

2 A new performance-centred maintenance (PCM) approach 26

2.1 Generic business setting and problem structure . . . . . . . . . . . . . . . 30

2.2 Conceptual framework, modelling framework and mathematical model . . 34

2.2.1 Conceptual framework . . . . . . . . . . . . . . . . . . . . . . . . . 35

2.2.2 Mechanism of the operators' decision regime . . . . . . . . . . . . . 38

2.2.3 Modelling the deterioration process . . . . . . . . . . . . . . . . . . 40

2.2.4 Modelling the maintenance planning optimisation problem . . . . . 41

2.2.4.1 Structure of the maintenance planning optimisation problem . . . . . . . . . . . . . . . . . . . . . . . . . . 42

2.2.4.2 Markov decision processes . . . . . . . . . . . . . . . . . . 43

2.2.4.3 Applying MDPs to our problem . . . . . . . . . . . . . . 45

2.3 Comparison with relevant studies . . . . . . . . . . . . . . . . . . . . . . 50

2.4 Necessity for heuristics . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51

3 Solving MDPs: computation methods and data-storage methods 56

3.1 Exact and heuristic computation methods (brief summary) . . . . . . . . . 57

3.2 Value iteration . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 59

3.2.1 VI for finite-horizon MDPs . . . . . . . . . . . . . . . . . . . . . . 59

3.2.2 VI for infinite-horizon MDPs . . . . . . . . . . . . . . . . . . . . . 60

3.2.3 Application issues of value iteration . . . . . . . . . . . . . . . . . 62

3.3 Q-learning . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 63

3.3.1 Q-learning for finite-horizon MDPs . . . . . . . . . . . . . . . . . . 64

3.3.2 Q-learning for infinite-horizon MDPs . . . . . . . . . . . . . . . . . 69

3.3.3 Further technical discussions . . . . . . . . . . . . . . . . . . . . . 72

3.4 Value function approximation . . . . . . . . . . . . . . . . . . . . . . . . 74

3.4.1 Different data-storage heuristic approaches . . . . . . . . . . . . . . 75

3.4.2 Different data-storage heuristic methods . . . . . . . . . . . . . . . 77

**4 Case study: modelling and heuristics 82**

4.1 Case study: problem structure and assumptions . . . . . 83

4.1.1 Background of the coal-fired power plant . . . . . 83

4.1.2 Operators' decision regime in power plant . . . . . 87

4.1.3 Hierarchical structure in general . . . . . 88

4.1.4 Case study . . . . . 90

4.2 Three-stage hybrid MDP model . . . . . 101

4.2.1 Decomposition method and modelling framework . . . . . 101

4.2.1.1 Two-timescale decomposition method . . . . . 102

4.2.1.2 Modelling framework . . . . . 106

4.2.2 Mathematical model . . . . . 107

4.2.2.1 MDP model for Stage (3) . . . . . 108

4.2.2.2 MDP model for Stage (2) . . . . . 111

4.2.2.3 MDP model for Stage (1) . . . . . 115

4.2.2.4 Interfaces between different stages . . . . . 118

4.2.2.5 Inputs and outputs of the model . . . . . 119

4.3 Heuristics for the three-stage hybrid MDP problem . . . . . 122

4.3.1 Polynomial function method . . . . . 123

4.3.2 Methods interaction and tuning free parameters . . . . . 132

4.4 Conclusion . . . . . 134

**5 Numerical tests 136**

5.1 Unit level: benchmarking heuristic methods . . . . . 137

5.1.1 Performance of heuristics and exact methods . . . . . 138

5.2 Plant level: comparison between different operators' decision regimes . . . 150

5.2.1 Numerical test setting . . . . . 150

5.2.2 Numerical results and discussion . . . . . 152

5.3 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 158

**6 Conclusions and further research 159**

6.1 Discussion on assumptions . . . . . . . . . . . . . . . . . . . . . . . . . . 163

6.2 Further research . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 168

**Bibliography 172**

**A Toy-size problem: compare different modelling approaches 191**

A.1 PCM approach: lists of additional parameters and derived decisions . . . . 191

A.2 Myopic approach: derived decisions comparison . . . . . . . . . . . . . . . 203

A.3 Perfect correlation approach: lists of additional parameters and derived decisions comparison . . . . . . . . . . . . . . . . . . . . . . . . . . . 211

A.4 Deterministic wearing-off approach: derived decisions comparison . . . . . 219

# Chapter 1

# Introduction

Machine maintenance is crucial to manufacturing industries to ensure the delivery of high-quality products to customers on time, and proper maintenance planning in general significantly contributes to a high level of operation efficiency [102]. Although the maintenance cost can range from 15% to 75% of the total operations expenditure in many cases and can even surpass annual net profit especially when no proper maintenance planning programs are implemented [99, 104], it is not an optimal choice for manufacturers to abandon machine maintenance at a strategic level [50]. In practice, manufacturers aim at deriving maintenance policies which render the trade-offs between various maintenance cost measures and various machine performance measures *approximately* optimal in their cases (note such trade-offs are usually very complex and therefore it is usually impossible to obtain exact optimal policies in practice). Motivated by such complex and important decision making problems about machine maintenance, researchers developed multiple maintenance approaches each of which emphasises on different crucial aspects of such trade-offs; the corresponding maintenance planning optimisation problems in studies/research are usually approximations of the actual problems in practice: a certain level of approximation of modelling is necessary as it enables researchers/practitioners to focus on the important features of the actual problems. Hereafter by *maintenance planning optimisation problems*, we mean such approximations rather than the exact problems in practice. Motivated by the case study of a real-life large scale power plant, this thesis focuses on a type of maintenance planning optimisation problem which emphasises the balance between machine utilisation and increased risk of failure.

Below first we shall present a brief literature review regarding existing maintenance approaches; then we shall explain the structure of the thesis and highlight the research questions and research contributions.

In order to facilitate the discussion in this thesis, here we distinguish the following three terms which shall be used repeatedly throughout this thesis: *maintenance approach*, *maintenance planning modelling framework* and *maintenance planning mathematical model.* Such three terms (or similar ones) have been repeatedly referred to in the machine maintenance planning literature; the definition of such terms however vary between different studies and the boundaries of such terms do not seem clear in literature, and hence here we re-define such terms to synchronise their boundaries and facilitate the discussion hereafter. In our thesis, a *maintenance approach* means a conceptual framework which specifies the fundamental trade-offs in maintenance planning decision making and the general principles about how the machine shall be maintained. Note such maintenance principles constrain the scope of potential maintenance policies in modelling. For instance, if a maintenance approach assumes the best way to maintain a machine in practice is to replace the old machine with a brand new one at fixed time intervals regardless of whether the machine still works and any intervening machine failure within a time interval must be immediately corrected with a machine replacement as well, then one would specifically focus on finding the time intervals which optimise some predetermined measures, e.g. the average maintenance cost over time. A *maintenance planning modelling framework* in our thesis refers to a framework which specifies how to mathematically model maintenance decision making given a *maintenance approach.* For example, one framework may choose a discrete-time setting whereas a different framework may adopt a continuous-time setting. A *maintenance planning mathematical model* in our thesis refers to a mathematical model which is derived through applying a *maintenance planning modelling framework* to maintenance planning decision making in a certain operations/business setting. For instance, in this thesis we construct a maintenance planning mathematical model for a case study from a real-life power plant. The relationship between the three terms is illustrated in Figure 1.1: a *maintenance planning mathematical model* must belong to a *maintenance planning modelling framework*; a *maintenance planning modelling framework* must belong to a *maintenance approach.* Hereafter we shall refer to *maintenance planning modelling framework* briefly as modelling framework and refer to *maintenance planning mathematical model* briefly as mathematical model, as long as this is clear in the context.

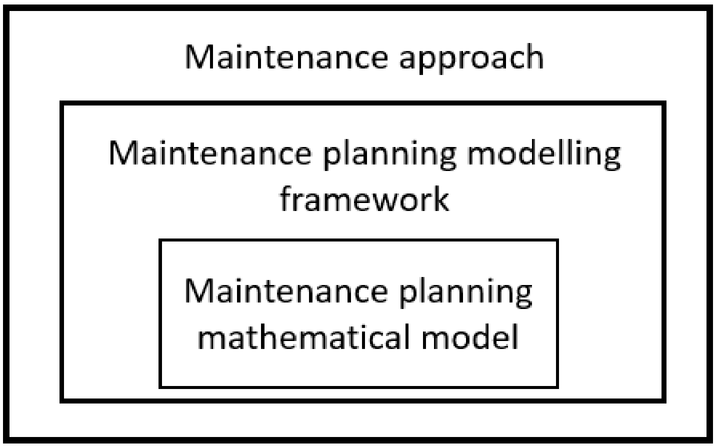

Maintenance approach
Maintenance planning modelling framework
Maintenance planning mathematical model

Figure 1.1: Relationship between maintenance approach, modelling framework and mathematical model

Regarding the three aforementioned concepts, from the maintenance approach to the modelling framework and finally to the mathematical model, inevitably more and more problem structure/modelling assumptions are introduced to facilitate the modelling work, especially if the maintenance planning decision making is complex in practice. These assumptions underpin the application scope of different maintenance approaches/ modelling frameworks/ mathematical models.

In the rest of the chapter, in order to maintain a relative high level overview, we shall focus on the conceptual frameworks of different maintenance approaches and the general categorisation of maintenance planning modelling frameworks, and we shall not discuss the maintenance planning mathematical models in the maintenance planning literature. This chapter aims at providing background knowledge for reader to understand the maintenance approach that we develop (in Chapter 2) and our research contributions to the literature.

## 1.1 An overview of machine maintenance approaches

Maintenance is a set of activities or tasks used to bring a machine back to a desirable state to provide required functions [49]. In this thesis, we use the term *machine* in a very general sense: it could either be a single piece of asset or a complex system which comprises of multiple assets, where each asset must be maintained in its entirety. We shall keep using such term to conduct the discussion in a general context, unless it is necessary to clarify whether we focus on a single asset or a multi-asset system with a specific physical structure.

In terms of the effects of a maintenance action, it often could be modelled as either *minimal*, *imperfect* or *perfect* [117]. A minimal maintenance action restores the machine to the failure rate it had when it failed: intuitively speaking, the machine is as bad as old after a minimal maintenance, for example changing a broken rubber belt on a flour mill would not improve the overall failure rate of the flour mill; a perfect maintenance action restores the machine to the state where it has the same lifetime distribution and failure rate as a brand new machine: intuitively speaking, the machine is as good as new after a perfect maintenance, equivalent to replacing the failed machine with a brand new one; the effects of an imperfect maintenance action fall somewhere between a minimal maintenance action and a perfect one: the machine after an imperfect maintenance becomes younger but not as good as new, for example the performance of an engine might be improved significantly after a tune-up but the machine deterioration is not reversed. Here the *failure rate* for a machine at a given time is defined as the ratio of dividing the probability that the machine of interest fails in an infinitely small time interval (following the given time) by the time interval itself, given that the machine is not failed at the given time [17]; the *machine failure* is defined as the machine of interest not being able to fulfil its required functions [46]: for example, if a printer cannot print materials with sharpness above an acceptable level, the printer is considered failed, even if it is not physically broken.

Other maintenance actions, in terms of their effects, include *worse maintenance* and *worst maintenance* [117]: a worse maintenance action would increase the machine failure rate and a worst maintenance action causes the machine to fail. Such maintenance actions are not a deliberate choice of the maintainers, but they might happen in practice due to multiple reasons including repairing the wrong part of the machine, wrong adjustments and replacement with faulty parts. We shall not consider worse maintenance and worst maintenance in our modelling work.

In terms of the timing of a maintenance action, it could be modelled as either *corrective* or *preventive*. Corrective maintenance (CM) actions refer to the maintenance actions executed after the machine fails, whereas preventive maintenance (PM) actions are the maintenance actions executed before the machine fails. Note that a CM action or a PM action can be minimal, imperfect or perfect regarding the effects [117].

Maintenance planning answers two questions: (1) when the machine should be maintained and (2) what maintenance actions should be selected (in terms of maintenance effects). The maintenance planning literature is rich in mathematical models which aim at facilitating optimal maintenance planning. Such (mathematical) models are built using a limited number of maintenance approaches and modelling frameworks. In this chapter, we shall focus on summarising maintenance approaches and modelling frameworks.

Maintenance approaches are generally classified as either corrective maintenance (CM) approaches or preventive maintenance (PM) approaches [53]. We shall first discuss the

CM approaches in Section 1.1.1, and then we focus on the PM approaches in Section 1.1.2. Additionally in Section 1.1.2 we shall also summarise the modelling frameworks in PM approaches because they directly relate to the later discussion in Chapter 2.

### 1.1.1 CM approaches

Maintainers who follow CM approaches restore a machine to its required functions after the machine fails [112]. In terms of the maintenance effects, the specific maintenance actions involved in CM approaches can be minimal, imperfect and/or perfect maintenance; in terms of maintenance timing, the specific maintenance actions involved in CM approaches are only CM actions. CM approaches are often intuitively known as run-to-failure maintenance approaches [2]. Such maintenance approaches may lead to a large amount of machine downtime, high maintenance costs and even serious safety issues to the personnel and environment due to a relatively large number of unexpected failures [50, 137].

[101] introduces the so called *repair number counting approach* which replaces the old machine with a new one (that is a perfect maintenance action) at the $k$-th failure and deals the first $(k-1)$ failures with minimal maintenance actions. The process repeats after replacement. [105] extends the approach by adding in another decision variable called critical reference time, denoted as $T$. The extended approach is called *reference time approach*. In this extended approach, all failures before the $k$-th failure are still corrected by minimal maintenance actions. Regarding the $k$-th failure, if it happens before an accumulated operating time $T$, it is corrected by a minimal maintenance action and the next failure is dealt with machine replacement; if the $k$-th failure happens after $T$, it is corrected by machine replacement. After machine replacement, the accumulated time $T$ is reset as zero and the process repeats. Readers are referred to [145] for a more thorough literature review on studies following this research line. Note in both CM and PM approaches, machine age is a virtual concept [90] and time is not always measured

as calender/clock time: for example in some studies the machine age is measured as the amount of products processed (see [15] for an example in the steel manufacturing industry) since the last perfect maintenance action, and such age is reset as zero after a perfect maintenance action.

Another CM maintenance approach is the so called *repair limit CM approach*. The approach applies minimal/imperfect maintenance when failure occurs, as long as the estimated maintenance costs/time-length is below a predetermined threshold; otherwise the the machine is replaced with a new one (that is a perfect maintenance action). Example studies include [51, 52, 108, 155]. Later on [87] extends the studies by additionally enforcing perfect maintenance after a certain number of minimal/imperfect maintenance; whereas [10] chooses to replace the one-time only minimal/imperfect maintenance cost (which is widely used in earlier studies) with repair cost rate (average repair cost per unit time) over a certain future planning period in the decision making.

### 1.1.2 PM approaches

In contrast to the CM approaches which only model interventions applied after a machine failure occurs, the PM approaches additionally model maintenance actions which are to be executed before the machine fails, in order to reduce the failure frequency of the machine in a finite or infinite planning period [109]. In terms of the maintenance effects, the specific maintenance actions involved in PM approaches can be minimal, imperfect and/or perfect maintenance; in terms of maintenance timing, the specific maintenance actions involved in PM approaches include CM actions and PM actions. PM approaches, when they are properly implemented in practice, reduce the total maintenance costs and machine downtime and improve the product quality [140], compared to CM approaches. The PM approaches are further classified as *time-based maintenance (TBM)* ones and *condition-based maintenance (CBM)* ones [3]. We shall first focus on the TBM ones and later on we discuss the CBM ones.

#### 1.1.2.1 TBM approaches

Maintainers who follow TBM approaches schedule the maintenance based on the priori statistical knowledge of the machine lifetime [61]. Their key goal is to derive the optimal time interval between every two successive preventive actions: as such time interval increases, the average cost of unplanned machine outage per time unit is assumed to increase whereas the average cost of preventive actions per time unit is assumed to decrease, and therefore a balance must be reached in order to achieve the minimum average cost per time unit [9]. According to [145], TBM approaches can be further categorised as the following: the *age-dependent PM approach*, the *periodic PM approach*, the *failure limit PM approach* and the *sequential PM approach*.

Following the *age-dependent PM approach*, the machine of interest receives a perfect PM action (usually it is machine replacement) once the the machine age reaches a predetermined threshold $T$ and any failures that occur before such an age threshold are corrected by CM actions. Depending on the modelling assumptions in specific studies, the CM actions can be minimal, imperfect and/or perfect maintenance actions. The accumulated machine age is reset as zero after a perfect maintenance action and the process repeats. For example, [9] only considers perfect maintenance for CM actions, whereas [106] additionally introduces an predefined threshold $N$ and [106] applies only minimal CM actions as long as the machine age is below $T$ and the failure number is below $N$; once the machine age reaches $T$ or failure number reaches $N$, whichever happens first, the machine receives a perfect maintenance and then the machine age and failure number are reset as zero.

Following the *periodic PM approach*, the machine of interest is preventively maintained at fixed (and usually equal) time intervals independent of the failure history of the machine and any intervening failures within an interval are corrected by CM actions. For example, [9] considers replacing the machine at predetermined identical time intervals and any intervening failures within an interval are removed by minimal repair; [107] focuses on an approach similar to [9], with one extra requirement on triggering the preventive replacement: the number of machine failures since the previous replacement must surpass a predetermined threshold, otherwise no replacement should be done; [146] discusses a maintenance approach similar to [107] but [146] considers preventive maintenance which is either perfect or minimal with fixed probabilities rather than preventive maintenance that is always perfect (and the accumulated number of failures is reset after a perfect maintenance in [146]).

Following the *failure limit PM approach*, the machine is only preventively maintained when the failure rate (or other reliability related measures, which we shall explain them later in this section) reaches a predetermined level and any intervening failures are corrected by CM actions. For instance, [94] introduces a maintenance approach which applies minimal PM to the machine whenever it reaches a predetermined failure rate threshold; [97] applies a failure limit approach under Weibull failure rates.

Following the *sequential PM approach*, it is not required to specify at the beginning of the planning period each future PM interval; instead, after each executed maintenance action, it is only required to specify the next PM interval. If the machine fails or the selected PM interval fully elapses, whichever happens first, a maintenance action is executed and the process repeats. The PM interval is selected based on the estimated remaining lifetime of the machine and cost structures. This approach aims at adding in flexibility compared to the *periodic PM approach* such that the approach can deliver better results under some predetermined measures (e.g. expected total maintenance costs). For example, [110] studies an sequential PM approach which faces a situation where each PM increases the failure rate of the machine; naturally the time interval for PM decreases as time passes until perfect maintenance is done; [91] considers a sequential PM approach in a cumulative damage shock environment where shocks occur following a Poisson process and the system fails with a probability depending on the accumulated damage due to the shocks.

Regarding all the aforementioned TBM approaches, the periodic PM approach is perhaps the easiest to implement in practice. The the age-dependent PM approach and sequential PM approach however are able to further reduce the maintenance cost, compared to the periodic PM approach. The failure limit PM approach is most suitable for maintenance planning decision making which focuses more on reliability (or reliability related measures) rather than cost, and we shall further discuss this issue in this section when we discuss optimisation objectives.

As pointed out by [145], other studies focus exclusively on maintenance planning for multi-asset machines where the assets are subject to dependencies including economic dependence (i.e. maintaining multiple assets simultaneously costs less money and/or time compared to maintaining them separately), failure dependence (i.e. the lifetime distributions of multiple assets are stochastically dependent) and structure dependence (i.e. the assets are bonded in a way such that maintaining any one asset pre-requires dismantling the assets apart from each other). Such studies focus on joint maintenance approaches including the group maintenance approach (i.e. a fixed category of assets are maintained jointly, for instance assets with the same lifetime distribution) and the opportunistic maintenance approach (i.e. when one asset receives maintenance, the other assets deemed to have little remaining lifetime are maintained as well). We refer readers to [92, 145] for a more detailed discussion.

**Modelling frameworks and optimisation objectives**

In the TBM approaches, the lifetime of a machine is modelled as a continuous random variable [147]. A key difference between various modelling frameworks in TBM approaches is how such random variable is modelled. According to [103], the distribution of the random variable is specified by one of several functions, including the *probability density function* (pdf), *characteristic function*, *Mellin transform*, *failure rate function*, *survival function* and *cumulative distribution function*. In order to decide the specific parameters of the corresponding statistical model such that the machine lifetime can be reasonably well modelled, techniques including goodness-for-fit tests and maximum likelihood estimation should be applied on the historical failure data and maintenance events data. We refer readers to [17] for detailed discussions. Regarding the issues in collecting and processing the historical failure data and maintenance events data, we refer readers to [2].

In the TBM approaches, a maintenance planning modelling framework specifies three things: the type of distribution regarding the machine lifetime variable (as explained above); the maintenance planning optimisation objective(s) (which is explained immediately below); function(s) that quantifies how much each decision variable (e.g. machine replacement age) contributes to the objective value(s).

Optimisation objectives

The optimisation objectives for maintenance planning include minimising some maintenance cost measures, maximising reliability and optimising some reliability related measures [145]. The maintenance cost measures include the maintenance cost rate (average maintenance cost per unit time), total maintenance cost and discounted maintenance cost; the reliability related measures include availability, failure rate and mean time between failures (MTBF) [145]: the *reliability* itself is the probability that the machine will not fail before it reaches a certain age [17]; the *availability* is the probability that the machine will be available when required, or the proportion of total time that the machine is available for use [113]; the *failure rate* is explained above already; *mean time between failures (MTBF)* applies to repairable machines and not to non-repairable machines.

According to [3, 145], the optimisation choices in publications can be summarised as five types: (1) minimising a maintenance cost measure; (2) maximising the reliability or optimising a reliability related measure; (3) optimising a maintenance cost measure while ensuring the requirement on the machine reliability or a reliability related measure is satisfied; (4) maximising the machine reliability or optimising a reliability related measure while ensuring the requirement on a cost measure is satisfied; (5) multi-objective optimisation which involve two or more conflicting objectives (for example minimising total maintenance cost and maximising reliability). The first choice is most common in the literature, because cost measures apply to many practical problem settings [86]; however, for problems in which the consequences regarding a low level of reliability (or its related measures) cannot be (fully) evaluated from the perspective of cost, sole cost-oriented optimisation should be replaced by other appropriate choices aforementioned [86, 145].

#### 1.1.2.2 CBM approaches

In the TBM approaches, the machine failure is assumed to be age-related [2], and the decision making is based on the machine life distribution rather than the actual state of the machine. As the modern production machines become more complex and the requirement on high machine reliability grows ever stronger, the maintenance cost in TBM approaches keep increasing. The maintenance engineers and researchers respond to such challenges with more sophisticated maintenance approaches: condition-based maintenance (CBM) approaches. Following CBM approaches, maintenance decision making is based on directly/indirectly monitored machine degradation condition [73], which reduces the uncertainty about time to machine failure: the monitoring data provides evidence or insight about whether the machine is working in an abnormal state and how serious it is. As a result, compared to TBM approaches, CBM approaches can reduce the maintenance cost and improve machine reliability by scheduling preventive maintenance actions more efficiently.

The relative effectiveness is dependent on problem-specific factors including the required setup time for desirable PM actions, the severity of failures, the accuracy of the condition measurements and the amount of randomness in the deterioration level at which failure occurs [44]. It is also worth mentioning that CBM approaches usually involve relatively high installation cost (e.g. sensors installation and staff training costs) in practice and we refer readers to [128] for detailed discussion regarding the advantages and disadvantages about CBM approaches.

The machine condition information is gather by using sensors and/or other appropriate proxies [24]. As summarised by [2], the machine condition monitoring techniques include vibration monitoring, sound or acoustic monitoring, oil-analysis or lubricant monitoring, electrical monitoring, temperature monitoring, physical condition (e.g. cracks and corrosion) monitoring, performance (e.g. flow rate and electrical power consumption) monitoring and so on. In this thesis, we focus on the maintenance decision making and therefore for CBM approaches we choose not to further discuss how the machine condition information is collected, the categorisation of inspection quality or how the inspection results are processed to reveal/infer the machine condition; instead we refer readers to [3, 86] for a thorough discussion. We shall assume the monitoring information provides an exact view of the machine state in our modelling work.

CBM approaches can be categorised based on the machine condition inspection frequency: *continuous monitoring*, *periodic inspection* and *non-periodic inspection* [86]. Continuous monitoring literally means the machine condition is monitored without any time gaps; periodic inspection means the machine condition is examined at fixed (and usually equal) time intervals; non-periodic inspection means after each inspection only the immediate next inspection interval is decided, and usually the inspection interval decreases as the machine deteriorates. The choice of the inspection intervals influences the performance of the ultimate maintenance policies, for example in terms of total costs and machine reliability. We refer readers to [86] for a detailed discussion regarding the advantages and disadvantages of different inspection frequency.

In CBM approaches which adopt periodic inspection or non-periodic inspection, the inspection frequency must be determined as part of the maintenance planning problem (the other issues have been discussed at the beginning of Section 1.1).

**Modelling frameworks and optimisation objectives**

In the CBM approaches, similar to the TBM approaches, a maintenance planning modelling framework specifies three things: how to model the stochastic machine degradation process; the maintenance planning optimisation objective(s); function(s) that quantifies how much each decision variable (e.g. inspection frequency or which machine states should trigger preventive maintenance actions) contributes to the objective value(s). Below we shall first focus on such different modelling frameworks regarding the machine degradation process and then we discuss the optimisation objectives.

A key difference between various modelling frameworks in CBM approaches is how the stochastic machine deterioration process is modelled. The deterioration process can be modelled from the perspective of discrete-state/continuous-state and discrete-time/continuous-time aspects.

Discrete-state deterioration modelling frameworks

As summarised by [3], discrete-state deterioration modelling frameworks include the *Markov chain*, *Semi-Markov process* and *hidden Markov process*. All the frameworks assume (1) the total number of potential machine states is finite and (2) the probability that the machine transits into a given state only depends on the current machine state and is independent from the historical machine states.

A *Markov chain* additionally assumes that the machine state transition always consumes a fixed time-unit; in other words, Markov chain takes a discrete-time setting for modelling deterioration and hence is also called discrete-time Markov process (or more specifically, discrete-time discrete-state Markov process) in some studies. Application studies of Markov chain in machine maintenance include [23, 55, 71, 88, 151, 152, 161].

In contrast, a *Semi-Markov process* assumes the state transition time is a continuous variable of which the distribution depends on the current state and the state to jump into and does not depend on any historical states; a special type of Semi-Markov processes further assumes the transition time is exponentially distributed and such Semi-Markov processes are called continuous-time Markov processes. Application studies of semi-Markov processes in machine maintenance include [27, 40, 41, 54, 74, 93, 102]; application studies of continuous-time Markov processes in machine maintenance include [30, 89, 124, 127].

The *hidden Markov process* is used when the deterioration dynamics are assumed to be determined by either a Markov chain or Semi-Markov process but the monitoring information cannot exactly reveal the machine condition; in order to handle such uncertainty the hidden Markov process additionally contains a vector of probability distributions where each potential machine state is described by a unique distribution regarding the probabilities of what kinds of output information are observable. The hidden Markov process can either take a continuous-time setting or a discrete-time setting to model the deterioration process. Application studies of hidden Markov process in machine maintenance include [25, 58, 96, 121, 156, 154].

A comparison summary is provided in Table 1.1 regarding the three deterioration modelling frameworks, as well as the associated maintenance planning modelling frameworks. Here we would like to highlight that the time-setting we discuss for each deterioration modelling framework above is specifically for modelling the deterioration process only: that is whether the machine state would change in a continuous-time setting or discrete-time setting due to deterioration, rather than for maintenance decision making; for example, the deterioration process of a machine is assumed to follow a continuous-time setting but the maintenance decision making can be following a discrete time-setting [98, 158]. In a maintenance planning modelling framework, if the deterioration process is assumed to follow a continuous-time setting, either a continuous time-setting or a discrete time-setting can be assumed for the maintenance decision making; if the deterioration process is assumed to follow a discrete-time setting, it is natural to only assume a discrete-time setting for the maintenance decision making. When we refer to time-setting, we would always specify whether it is a time-setting for deterioration or it is for inspection and decision making.

Regarding all the discrete-state deterioration modelling frameworks aforementioned, the Semi-Markov process and hidden Markov process are more general frameworks than the Markov chain, but they also require more data collection and data processing and

| **Deterioration modelling framework** | **Deterioration time-setting** | **Corresponding maintenance planning modelling framework** |
|---|---|---|
| Markov chain | Discrete | Markov decision process (MDP): discrete-time setting for decision making |
| Semi-Markov process and continuous-time Markov process | Continuous | Semi-Markov decision process (Semi-MDP) and continuous-time Markov decision process (CTMDP): continuous-time/discrete-time setting for decision making |
| Hidden Markov process | Discrete/continuous | Partially observable Markov decision process (POMDP): continuous-time/discrete-time setting for decision making |

Table 1.1: Comparison between discrete-state deterioration modelling frameworks

statistical analysis. In comparison to the continuous-state deterioration modelling frameworks to be discussed below, the discrete-state deterioration modelling frameworks are usually used either when precise measurements of the machine states cannot be obtained or as an approximation of the actual degradation process from an engineering practice perspective (i.e. categorising the degradation states into several deterioration levels) [3].

Proportional hazard modelling (PHM) framework

Used in CBM approaches, the proportional hazard model [43] describes the machine state by its failure rate and assumes the failure rate of the machine at any given time is influenced (i.e. accelerated/decelerated) by multiple factors that can be continuously monitored, e.g. temperature and running speed [3]. Such factors can either be condition-based or external to the machine. Technically speaking, a proportional hazard model describes the failure rate of a machine as the product of two failure rate functions: one baseline failure rate function of time; one failure rate function of the measurement values of the influencing factor at the given time [103].

Compared to other deterioration modelling frameworks, the PHM framework is most suitable for maintenance planning decision making which focuses more on reliability (or reliability related measures) rather than cost, such as the maintenance planning for critical equipments in nuclear power plants.

Additional deterioration modelling frameworks

According to [3], some other frameworks model the machine deterioration through time as a continuous-state process rather than transitions between discrete states; such frameworks include the *Wiener process*, *Gamma process* and *inverse Gaussian process*. All such frameworks take a continuous-time setting for modelling deterioration.

The *Wiener process* assumes the deterioration is non-monotonic over time, and it is often used to model the deterioration process which shows a mixture of increments and decrements of degradation over time: for instance the performance degradation of self-regulating heating cables [150] and semiconductor laser devices [160]. Technically speaking, the Wiener process is a stochastic process with independent and normally distributed increments or decrements [153].

In contrast, both the *Gamma process* and the *inverse Gaussian process* assume the deterioration is monotonic over time, and they are applied to modelling the deterioration process which takes the form of cumulative damage: that is irreversible deterioration. Technically speaking, the Gamma process is a stochastic process with independent and Gamma-distributed increments [3]; the inverse Gaussian process is a stochastic process with independent and inverse Gaussian distributed increments [3]. Because the Gamma process has only one way of mathematically incorporating the random effects from external shocks on the deterioration process while the inverse Gaussian process has three ways (all different from the way in the Gamma process) of doing so [153], researchers see the inverse Gaussian process as a more flexible modelling approach than the Gamma process.

Due to the specific statistical assumptions embedded in each modelling framework above, goodness-for-fit tests should be applied in order to decide whether a modelling framework is suitable or which modelling framework is the best to model the degradation process of interest. We refer readers to [153, 157] for detailed discussions.

In most studies which adopt continuous-state deterioration modelling frameworks, the

CBM approaches maintain the machine by exclusively following the *control-limit policies* [2, 3, 86, 137, 147]. In a control-limit policy, the preventive maintenance actions are triggered if and only if the detected machine condition deteriorates to or worse than a predetermined level [81].

Optimisation objectives

In the TBM approaches, only two states of the machine are modelled in general: failure and non-failure. In the CBM approaches, researchers recognise that a machine can potentially go through multiple operational states before it fails. In other words, the objective values aforementioned in Section 1.1.2.1 in TBM approaches are measured based on statistical results that reflect "average" reliability characteristics [147]. Here we discuss how such measuring should be updated for the CBM approaches.

Regarding the maintenance cost measures (i.e. maintenance cost rate, total maintenance cost and discounted maintenance cost), they should be evaluated based on the maintenance cost associated with multiple operational states that a machine can potentially evolve into at different stages throughout a planning period, rather than merely the maintenance cost associated with two states (failure and non-failure) [84, 93, 141, 142].

Regarding reliability and reliability-related measures, their values should be evaluated as follows: *reliability* should be measured as the probability that the machine is in an operational state (rather than *the* single non-failed state) before it reaches a certain age [127]; *availability* should be measured as the probability that the machine is in an operational state (rather than *the* single non-failed state) when required [93]; *failure rate* should be measured under the additional prerequisite that it is known which operational state the machine is in at the given time [147]; *mean time between failures (MTBF)* should be measured as the mean time that the machine spends at the operational states (rather than *the* single non-failed state) before it fails [102].

We highlight the maintenance approaches discussed in this chapter in Figure 1.2. As illustrated so far in this chapter, the advancement of maintenance approaches in literature contributes to more complex mathematical modelling work which aims at better capturing ever-improving (that is more comprehensive and accurate) monitoring information on machine state in practice, which as a result facilitates more effective maintenance planning decision making from the perspective of managing increased risk; however, the literature so far largely overlooked another crucial aspect regarding maintenance planning decision making in practice: machine utilisation. We shall discuss such issue in details in Chapter 2.

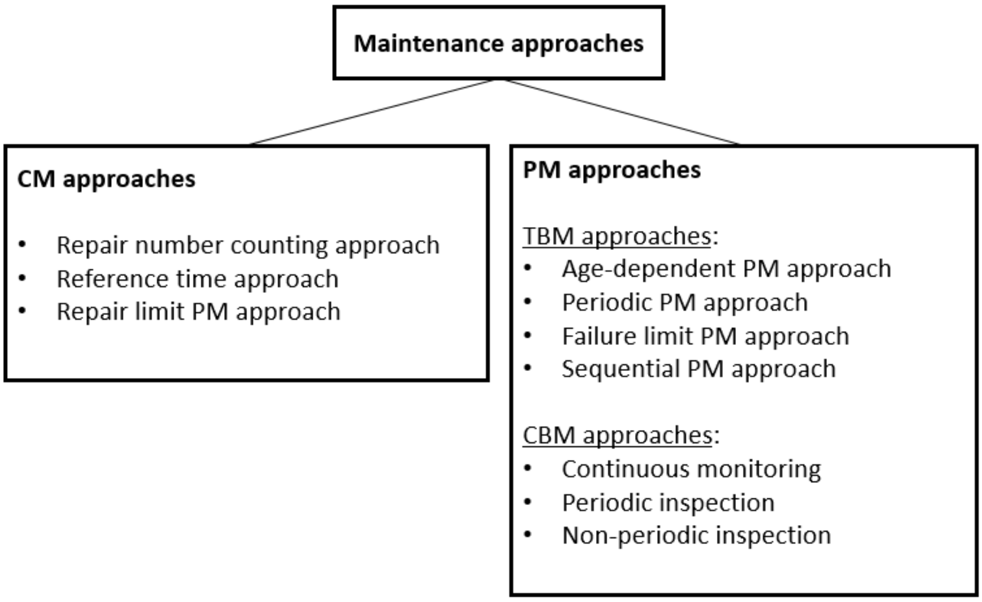


Figure 1.2: Different maintenance approaches

## 1.2 Further literature review on relevant studies

In the research area of machine maintenance approaches, two specific research domains contain perhaps the most relevant studies to our research: (1) studies that model machine operational performance and (2) studies on maintenance-production joint scheduling. Here we shall present a brief literature review on some representative studies and highlight their common limitations from the perspective of our maintenance approach (our maintenance approach shall be specified in Chapter 2).

### 1.2.1 CBM studies involving operational performance

Before introducing studies that model machine operational performance, here we specify two terms: *condition* and *performance*. According to [8] , the *condition* and *performance* of a machine is each comprised of multiple *variables/measures*. A *condition variable* is some measurement of the indicator of a potential failure mode of an asset of interest, for example wear; a *performance variable* is some measurement related to the quality/quantity of the asset's production output, for example alignment. Each degradation condition measure and operational performance measure is associated with a threshold level and the machine fails if any condition/performance measure deteriorates to the corresponding threshold level.

From the perspective of our maintenance approach, most condition-based maintenance (CBM) planning studies (for instance [29, 41, 93, 120, 126, 147]) do not involve the concept of operational performance at all. These studies are not part of our discussion here; instead, here we focus on CBM planning studies from other research lines that do involve the concept of operational performance. Compared to our research, these condition-based maintenance studies do not consider the situations where condition and performance may not be perfectly correlated and some maintenance actions may not be equally effective on improving both condition and performance. Additionally, these studies lack the consideration about impacts on the maintenance planning from operators' decision regimes.

In some of such studies, performance-monitoring acts as a surrogate measure for the degradation condition of the asset of interest, rather than a separate entity to degradation condition. [14] discusses monitoring abnormalities on certain types of performance indicators of wind turbines and how to use the abnormalities to detect pollution and malfunction of some key components in electrical pitch systems of wind turbine generators before critical faults occur. [159] focus on applying a structural health monitoring system on arch bridges such that an array of sensors can periodically measure things including the structural performance which are used to infer the deterioration or damage of the arch bridges at an early stage. Another example [16] studies how to infer the degradation and predict faults of the semiconductor manufacturing systems based on measuring a large set of performance indicators.

Some other studies are dedicated to addressing operational performance deterioration but they do not discuss degradation condition. For example, [139] addresses the issues of water filtration performance monitoring and assessment, as well as preventive maintenance planning to improve poor filtration performance for rapid gravity filters; [115] discusses how to define the threshold values for pavement surface characteristics (including skid resistance, evenness and rutting) and argue that maintenance should be initialised once measurements of all three characteristics drop below corresponding thresholds; [57] studies how to improve the maintenance planning of so-called ground track cross node in order to ensure the navigation performance of the inclined geosynchronous orbit satellites.

Some studies address degradation condition and operational performance as two different entities, but they assume perfect correlation exists between them throughout the lifetime of an asset. [46] considers optimising predictive maintenance policies for a gradually deteriorating single-asset system which is subject to stress, and the operational performance failure is assumed to be always caused by asset deterioration beyond a certain threshold; [127] develops the optimal maintenance policy for a system which can potentially go through multiple ranked intermediate operational performance states and ranked degradation condition states before the system fails, where each performance state is assumed exclusively associated with a distinct condition state. [19] optimises the maintenance strategy in a case study about offshore wind farms in order to minimise the maintenance cost and maximise the equipment availability and energy production performance, where the performance indicator is the total produced energy which directly depends on the amount of equipment down time.

### 1.2.2 Maintenance-production joint scheduling

The other research area we look into consists of the maintenance-production joint scheduling studies. Similar to our research, such studies address the issue of unifying the production planning and maintenance planning for better costs/profits. More specifically speaking, such studies in general aim at planning maintenance and production jointly up to a certain time horizon such that the deterministic time-dependent forecasted demands are optimally fulfilled in terms of the trade-offs between costs and benefits including the production cost, (preventive and corrective) maintenance cost, inventory cost and sales revenue. Compared to our research, these studies however assume perfect correlations between the operational performance and degradation condition (or even only consider one of them) as well as deterministic numerical relationship regarding the dependence of assets deterioration on the production, which excludes the realistic possibilities of imperfect correlation between the performance and condition (as discussed in the beginning of Chapter 2) as well as the uncertainties involved in decision makings discussed in Section 2.2.4.1.

Some earlier maintenance-production joint scheduling studies are overly production focused: they require the maintenance scheduling is somehow known and fixed in advance (see [1, 31, 45] as examples). Such studies are not part of our discussion here; instead, here we are interested in the more balanced studies that consider the dependence of maintenance requirements on the production and do not assume pre-fixed maintenance or production planning. Some of such studies consider the deterministic impacts on condition degradation from production (see [18, 28, 76, 111, 144] for examples) and they do not involve the concept of operational performance.

Some other studies consider the performance decaying of assets with production time: [85] tackles the cyclic scheduling problem for continuous parallel-process assets and two operation modes are considered where the conversion rate for the key product or processing rate of input materials decreases exponentially with the production time since the last cleaning activity. [5] extends the research into multi-product multi-stage plants. [6] considers an economic lot scheduling problem similar to [85] with extra consideration of inventory costs, assuming the yield of the key product decays linearly with time. Other studies along this research line include [26, 59, 60, 77, 78, 79, 95, 125]. These studies assume the performance of an asset is fully restored after a cleaning maintenance. Degradation condition is however excluded from these studies and the machine deterioration processes are assumed to be deterministic.

[15] considers the impacts of production on the deterioration of both condition and performance in a flow-shop system: the production is assumed to imposes deterministic wearing effects on the residual lifetimes of the assets in a linear way which is specified under each type of production task and operation mode for each asset; [15] further assumes certain measures of the production performance decreases deterministically with the residual lifetime of an asset (the maximum production batch size that an asset can handle linearly depends on its residual lifetime and the availability of each operation mode of an asset is associated with a certain threshold requirement on the asset's residual lifetime), and an asset is assumed to be restored as good as new after a maintenance. The mathematical model in such study is exclusively based on deterministic deterioration assumptions, and performance is assumed perfectly correlated with the condition.

## 1.3 Thesis structure and research objectives

In Chapter 2, motivated by the generic maintenance planning problem extracted from a large scale power plant case, we shall thoroughly investigate the following research question: *how to balance between machine utilisation and increased risk of failure in a business setting where (1) machine maintenance and machine utilisation are tightly intertwined and (2) managerial parties follow different decision making time-scales?* As we shall specify in Chapter 2, such research question is common in the manufacturing industry and it cannot be effectively resolved by existing maintenance approaches. In order to fill in such an important research gap between the maintenance approaches literature and the research question of interest, in Chapter 2 we shall develop a new maintenance approach which (1) captures how the decision making time-scales of different managerial parties impact their value-perception of various operations activities and (2) facilitates operations decision making from a balanced view between machine utilisation and increased risk of failure.

The application of such maintenance approach in industrial cases results in large size mathematical problems which cannot be effectively solved by brute-force methods, which gives rise to the second research question we shall investigate in this thesis: *how to effectively solve such large size mathematical problems which would induce (1) impractically long computational time and (2) impractically large data-storage cost if brute-force methods are applied?* First, in Section 2.4 we shall explain three measures we refer to when evaluating the effectiveness of different methods: computational time, data-storage cost and data accuracy level; additionally, in Section 2.4 we shall also specify the numerical benchmark for each such measure. Then in Chapter 3 we shall specify some existing heuristics and justify our selection of such heuristics, and finally based on such selected existing heuristics we shall in Chapter 4 design our own heuristics in order to effectively solve the large size mathematical problems of interest.

Additionally, as we shall discuss in Chapter 4, further investigation of the specific power plant case study background highlights a general *hierarchical physical structure* that requires maintenance resources distribution at multiple sequential levels and such structure is common in industries which have adopted distributed generation/production. The hierarchical structure is beyond the scope of the mathematical model developed in Chapter 2 and therefore raises the third research question in this thesis: *how to scale up the mathematical model from Chapter 2 to facilitate maintenance planning optimisation in a more complex business setting which further involves the aforementioned hierarchical* *structure?* In response, we shall in Chapter 4 build a more sophisticated new mathematical model following the maintenance approach from Chapter 2 in order to capture optimal maintenance planning decision making for more complex industrial problems that further involve the hierarchical structure. Note that in Chapter 4 first we shall develop the new mathematical model and the aforementioned heuristics in the context of the specific power plant case and then we shall discuss how such mathematical model and heuristics can be applied to other cases which also follow the *hierarchical physical structure.*

In Chapter 5, we shall present numerical test results in the context of the power plant case in order to demonstrate the practical value of our maintenance approach, mathematical model and heuristics. In Chapter 6 we shall summarise our study and highlight some future research directions.

In summary, the main research contributions of the thesis are as follows:

- Chapter 2: building a maintenance approach to balance between machine utilisation and increased machine failure in a relatively generic manufacturing problem setting where machine maintenance and machine utilisation are tightly interwind and decision makers follow different time-scales in their operations decision making.

- Chapter 4: developing a mathematical model to capture optimal maintenance planning decision making for more complex industrial problems which further involve the hierarchical structure of interest.

- Chapter 4: constructing a set of heuristics to effectively solve the large size mathematical problems resulted from applying our maintenance approach to industrial cases, including the industrial cases which further involve the aforementioned hierarchical structure.

# Chapter 2

# A new performance-centred maintenance (PCM) approach

The recent research [7, 8] of the maintenance planning problem in a power plant highlights two general issues which challenge the fundamental conceptual frameworks of the existing maintenance approaches summarised in Chapter 1. Below we discuss the two issues in a general production-maintenance context rather than merely in the specific power plant context.

In practice, the machine maintainers and machine operators manage the operations of their production system together, sharing a responsibility of generating profits by fulfilling the demand of production output with a relatively low operational cost. The operations management is usually split between the two managerial parties as follows: the operators plan and execute the production and the maintainers schedule and apply the maintenance. Such a differentiation of business functions render the focus of the two parties different: the operators focus on meeting the demand while the maintainers focus on maintaining the machine health. Despite the difference in focus between the two management parties, the maintenance operations management and production operations management are intertwined with each other in practice: the decision making of the two managerial parties impacts each other in reality via directly influencing the machine state, which the two types of decision making themselves depend on in turn. More specifically speaking, a scheduled maintenance action would improve the machine state which leads to better production performance (in terms of quantify/quality) in the future for operators, but maintenance also induces potential loss of production during maintenance (due to pulling off machines that can still work) for operators as well as certain fixed costs; on the other hand, the operators may sometimes decide to speed up production in order to meet the demand but such machine utilisation behaviour would make the machine state deteriorate faster and hence induces higher necessity for maintenance in the future (note hereafter by "intertwined" we refer to such impacts between the decision making of operators and maintainers induced by the mutual dependence between decision making and machine state). Therefore, maintenance operations management and production operations management should not be considered independently from each other. In the maintenance planning literature, however, the balance between machine utilisation and increased risk of failure is rarely thoroughly discussed.

Moreover, in line with the widespread installation of sensors into various types of industrial production systems over the recent years, as explained in Section 1.1.2.2, the maintenance planning literature is shifting from the time-based maintenance (TBM) approaches which plan preventive maintenance actions based on a priori statistical knowledge of the system lifetime to the machine condition-based maintenance (CBM) approaches which plan preventive maintenance actions based on monitoring the degradation condition of the machine. CBM approaches allow for more effectively maintenance planning by reducing the uncertainty about time to machine failure. But so far in the maintenance planning literature, including the CBM planning studies, the operational performance of the machine is still largely bounded by the assumption of a perfect link with the degradation condition. In reality, take the power plant which motivates our study for example, although the condition deterioration underpins the performance deterioration, the perfect link in assumption does not necessarily apply. In contrast, in practice maintainers need to prioritise between maintenance alternatives that have a crucial impact on the performance but little effects on the condition and other maintenance alternatives that have significant effects on the condition. The former alternatives may be beneficial from the viewpoint of improving short-term production performance with relatively low fixed maintenance costs and relatively short maintenance times, but they have very little effects on improving the long-term reliability of the machine; the latter maintenance alternatives can improve long-term reliability, but they usually have higher fixed costs and consume longer times which potentially induces higher interventions in production.

The two issues aforementioned regarding the maintenance planning literature imply the following fundamental changes to the lifetime/condition-centred conceptual framework which underpins most of the maintenance planning optimisation studies so far: the conceptual framework should enlarge its scope to include not only machine reliability, but also the potential maintenance interventions in production and the potential impacts on maintenance planning from different machine utilisation behaviours, and a reasonable approach to capture such impacts between the decision making of maintainers and operators is to investigate and model the aforementioned mutual dependence between the decision making and machine state; additionally, the framework should consider operational performance and degradation condition as two separate entities. The resulted maintenance approach is potentially sophisticated and it enables both the maintainers and operators to align their business functions with the higher level of responsibility which is shared between the two managerial parties. Such insight motivates the proposal and development of the so called *performance-centred maintenance (PCM) approach* in [7, 8], as opposed to the lifetime/condition-centred maintenance approaches.

Our study follows the PCM research line and we focus on two important new issues that are not effectively resolved in the previous studies: (1) the machine operators and maintainers have different time-scales in practice (we shall explain it in Section 2.1) regarding their operations decision making, and such time-scale distinction modifies the difference between the two managerial parties regarding how they perceive the value of various operations activities (e.g. schedule a performance-oriented maintenance action, schedule a condition-oriented maintenance action or speed up the production for a certain time period). The larger the time-scale distinction is, the bigger the value-perception difference becomes between the two operations managerial parties. The previous studies do not clearly discuss such issue. In this study, we shall thoroughly investigate such issue and develop a new maintenance approach based on the existing PCM approach, in order to better support the decision making of both managerial parties and align their business functions more effectively. (2) The resulted mathematical models from applying the PCM approach to industrial cases are too complex to be solved by exact methods, and therefore heuristic methods are required instead. In this study we shall design a set of heuristics in response. (Part of the research results originated from this study regarding heuristics design and heuristics application has been combined into [8]; such research results are relatively early-stage output and in this thesis we shall deliver a much more comprehensive and advanced work based on our updated research results.)

The rest of the chapter is dedicated to issue (1) above, i.e. developing a new PCM approach to incorporate the time-scale distinction between the operators and maintainers; issue (2) is resolved in Chapter 3 and Chapter 4.

Below we first describe the general business setting of interest in Section 2.1; then we build the aforementioned new PCM approach in Section 2.2: more specifically speaking, in Section 2.2 we shall build a new conceptual framework, specify the modelling framework and develop a general mathematical model. Next, in Section 1.2 we compare the PCM approach with other relevant studies to further highlight the importance and novel aspects of the research line this thesis follows. Finally, in Section 2.4 we summarise our main research contribution of developing the new PCM approach and we discuss the general computation difficulties of solving the mathematical problems that result from applying the PCM approach to specific industrial cases, which lays the groundwork for discussing heuristics in Chapter 3-4.

## 2.1 Generic business setting and problem structure

Motivated by a real-life coal-fired power plant, this study considers a generic production system which contains multiple individual assets working in parallel, where the output quantity of the system is determined by both the quantity and quality of material processing of the individual assets. Product storage is either impossible or very expensive such that inventory would never be an optimal choice and therefore it is not considered. For instance, in a coal-fired power plant, every electricity production unit contains multiple mills. The mills grind coal into combustible dust which is to be burnt and converted into electricity. As a result, the electricity generation quantity depends on how much coal is ground and how fine the grinding is. The electricity cannot be stored on a large scale given the current technology [135]. We would like to highlight that the discussion in this chapter is dedicated to a general business context, and we use the power plant as a specific illustrative example.

The generic system faces an exogenous demand, and the sales of the output are contracted for a relative short-term (referred to as the *contracted period* hereafter) in advance in terms of both quantity and sales price. The system deteriorates with production and its state only improves after maintenance. The maintenance and operation of the system are carried out by two parties: the maintainers and the operators respectively. The maintainers undertake a business function of scheduling and applying the maintenance and their focus is to ensure that the system is available as required for production in a relative long term. The operators undertake a business function of planning and executing the production and they are highly motivated to fulfil the contracted demand, because missing contracted demand triggers a potentially high penalty cost. For instance, in the coal-fired power plant case study, if the production cannot satisfy the contracted demand, in order to resolve the shortfalls the plant owner must procure an emergent supply from the spot market but it may be much more expensive compared to self-production. The premium on the spot market is one example of the penalty cost. Shortfalls happen mostly due to unexpected failure or bad performance of the system which deteriorates stochastically. In practice, to avoid/mitigate the potential penalty cost, the operator may choose to speed up the production, if the system cannot fulfil the contracted demand by working at a normal rate. Such operation may be beneficial in the short-term contracted period, but it speeds up the deterioration of the system and therefore sacrifices the health of the system in the long-term.

Due to the difference regarding the business functions between the two managerial parties, the operators and maintainers perceive the value of various operations activities differently: the operators would judge the value of scheduled operations activities based on whether they enable the system to meet the production targets in the production planning period, whereas the maintainers would judge the value of scheduled operations activities based on their impact on the continuation of a relative good machine health in the maintenance planning period. Such two value perception perspectives are somehow both compatible and conflicting: a scheduled maintenance action would improve the machine state (which is favoured by the maintainers) and therefore contributes to better production performance (in terms of quantify/quality) in the future (which is favoured by the operators), but maintenance also induces certain fixed costs as well as potential loss of production during maintenance (which operators dislike); on the other hand, scheduling higher than normal production rates may help meet the demand (which is favoured by the operators) but such machine utilisation behaviours would make the machine deteriorate faster (which the maintainers dislike). The compatibility/conflict level regarding the value perception perspectives between maintainers and operators is modified by the distinction between the time-scales of the production planning period and the maintenance planning period, which we shall specify immediately below.

The existence of the contracted period for sales (and the associated potentially high penalty cost, which is an embedded characteristic of the contracted period in practice) naturally renders the operators more short-term oriented compared to the maintainers in terms of their operations decision making. In other words, the production planning period is shorter than the maintenance planning period due to the contracted period. Additionally, given that inventory is not a optimal choice in the business setting, the operators would have little motivation to plan beyond the contracted period. The shorter the production planning period is, compared to the maintenance planning period, the more conflicting rather than compatible the two value perception perspectives aforementioned become: the operators are cautious about the continuation of a relative good machine health in the production planning period because such continuation is preliminary to meet the production targets, but the shorter the production planning period is the weaker such caution becomes since the machine is less likely to degrade to a relatively bad state in a shorter production planning period. Therefore the shorter the production planning period is (compared to the maintenance planning period), the operators would be more likely to ignore the benefits of machine state improvement and see maintenance as a disruption to production, and meanwhile the operators would be more motivated to speed up production to resolve potential production shortfalls. In summary, the gap between the value perception perspectives of the two managerial parties is modified by the distinction between the time-scales of the two managerial parties in their decision making, and the contracted period plays a crucial role in such a time-scale distinction.

Given such important impact of the contracted period on modifying the gap between the value perception perspectives of the two managerial parties, it is necessary to reconstruct the conceptual and modelling frameworks of the existing performance-centred maintenance (PCM) approach in order to properly capture the effects that the contracted period has on shaping how the operators perceive the trade-offs differently than the maintainers in their decision making.

Having the contracted period as a key modelling aspect in our study makes our work unique compared to some earlier relevant research we build on: [7] models the maintenance planning problem for the specific power plant and to our knowledge they firstly introduced the terminology *performance-centred maintenance (PCM) approach* ; [8] generalises the modelling work of [7] for a broader problem setting and explicitly incorporates the operators' decision making in their conceptual framework. Both [7, 8] however do not capture the modifying effects of the contracted period on the gap between the operators and maintainers regarding their value perception perspectives of various operations activities (note although [8] captures the operators' decision making in their conceptual framework, they do not consider that the contracted period is usually shorter than the maintenance planning period, and in their mathematical model the demand is assumed to be contracted up to the end of maintenance planning period). Our study fills in such an important research gap: we reconstruct the PCM approach and underpin it with the concept of contracted period (we shall specify the modelling work in Section 2.2.1).

Given the complex relationship between the value perception perspectives of the two managerial parties, the maintenance planning is challenging in such a business setting: first of all, the system contains multiple assets each of which further has degradation condition and operational performance that are not necessarily perfectly correlated, which implies a wide range of maintenance alternatives as the maintainers prioritise different assets for either condition or performance focused maintenance, where each maintenance alternative represents a unique trade-off between a fixed maintenance cost and future rewards. Secondly, the maintainers need to consider potential loss of production during maintenance: an asset does not contribute any production to the system when it is taken off line for maintenance. In more details, regarding any maintenance actions scheduled within the contracted period, the maintainers need to consider the possibility of penalty cost induced by the potential loss of production during maintenance; as for any maintenance actions scheduled after the contracted period, the maintainers need to consider the possibility of loss of sales caused by the potential loss of production: unsatisfied demand would be lost and does not generate any revenue. Thirdly, the maintainers need to consider that the operators may speed up production in the contracted period and therefore render machine deterioration faster, which potentially induces higher necessity for maintenance in the long-term. Additionally, the uncertainties of a random demand after the contracted period and a stochastic machine deterioration process further complicate the analysis. In summary, the reconstructed PCM approach needs to consider complex uncertainties and consequences of a wide range of maintenance alternatives, as well as the impact of the operators' behaviours on maintenance planning. Therefore the reconstructed PCM approach is potentially sophisticated in order to balance off machine use against increased risk of machine failure and obtain the optimal trade-off between costs and benefits.

It is potentially complex to accurately model the maintenance planning optimisation problem in such a business setting. In this chapter, we shall therefore introduce some simplification assumptions and modelling choices in Section 2.2 in order to focus on the main features of the problem. The conceptual framework and the modelling framework that we shall reconstruct below for the PCM approach are based on such assumptions and modelling choices. In Chapter 6 we shall further discuss how to change such assumptions in future studies in order to develop more accurate mathematical models.

## 2.2 Conceptual framework, modelling framework and mathematical model

Below we first reconstruct the conceptual framework in Section 2.2.1 for the PCM approach, and then we decompose the discussion of reconstructing the modelling framework for the PCM approach as follows: in Section 2.2.2 we explain how to model the machine utilisation behaviours of the operators; next, in Section 2.2.3 we explain how to model the machine deterioration process; finally, in Section 2.2.4 we discuss how to model the maintenance planning optimisation problem.

### 2.2.1 Conceptual framework

In this section, we first introduce some simplification assumptions regarding the business setting and then we reconstruct the conceptual framework of the PCM approach based on the existing work from [8]. The key difference is we reconstruct the conceptual framework to capture how the contracted period modifies the perception of relevant trade-offs in the production and maintenance planning decision making.

We assume the following: (1) the machine state is continuously monitored and the monitoring information can accurately reveal the machine state; (2) the prices are all fixed throughout time such that the penalty cost and the profit per unit of demand are constants; (3) the contracted period has a fixed time length; (4) the demand after the contracted period is independent and identical distributed (i.i.d.), and we further assume that the distribution and its parameters can be reasonably inferred (e.g. based on market research or historical business data); (5) discrete time-setting for both the machine deterioration process and the decision making. The prices in reality actually evolve stochastically, and we may model the prices from the perspective of stochastic processes in future studies.

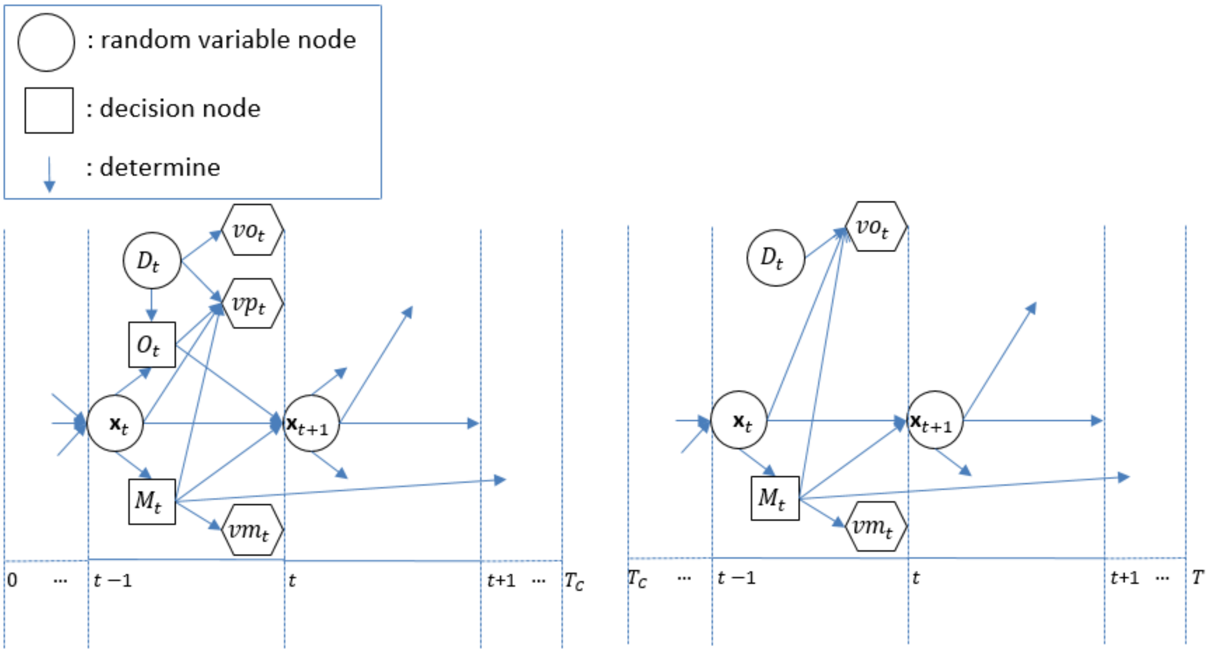
The maintainers need to plan maintenance for the system up to a certain horizon. The demand of the output is contracted for a short-term but it remains unknown after the contracted period up to the maintenance planning-horizon. The system state evolves stochastically and it can be revealed by sensor data. Each scheduled maintenance action is supposedly to improve the future state of the system; more specifically speaking, each maintenance action addresses either the condition and/or performance of a targeted asset in the system and each maintenance action has its own fixed time length. Any maintenance action has its own fixed cost, and it may further trigger either a penalty cost or sales loss, depending on whether the maintenance action is scheduled in the contracted period or after. Furthermore, in the contracted period, the operator attempts to meet the pre-agreed demand to create a revenue and the machine utilisation behaviour of the operator may influence the future state of the system. In addition, the decision making of the maintainers is subject to certain maintenance resource constraints which limit the availability of potential maintenance alternatives. In this chapter we use the term *maintenance resource constraints* in a very general sense for the conceptual framework and modelling framework that we shall build, and such constraints can only be specified in specific case studies (for instance the power plant case study in Chapter 4).

Here we accordingly introduce some mathematical notations: the time-length of the maintenance planning period is $T$; the time length of the contracted period is $T_C$, and $T_C < T$; the system state at time-step $t$ is denoted as $\mathbf{x}_t$, which is a vector comprised of the state of every asset in the system, and the state of an asset is further modelled by two separate entities: degradation condition and operational performance, each of which could further be comprised of multiple measures; the demand at time-step $t$ is denoted as $D(t)$, and $D(t)$ is deterministic for $t \leq T_C$ and stochastic for $t > T_C$; the total sales revenue at time-step $t$ is denoted as $vo(t)$ and the total penalty cost is denoted as $vp(t)$; the maintenance cost at time-step $t$ is denoted as $vm(t)$.

The conceptual relationship between the variables is illustrated in Figure 2.1 (a) and Figure 2.1 (b) each of which shows the operators' decision $O(t)$ and maintainers' decision $M(t)$ as well as the other variables discussed above, in the contracted period and after the contracted period respectively. Note that the operators' decision $O(t)$ only shows up in the contracted period because the operators are highly motivated to adjust the machine utilisation behaviour in order to fulfil the contracted demand; beyond the contracted period, the demand is not agreed yet and we choose to plan maintenance based on the premise/assumption of a normal machine utilisation pattern after the contracted period (note this assumption helps simplify the resulted mathematical model by avoiding modelling the consequences of different utilisation patterns with demand uncertainties; future research can choose to change such an assumption and derive more complex and accurate models). The approximation effects of such an assumption is mitigated by the rolling horizon nature (which we will illustrate in Section 2.2.4.3) of our maintenance approach.

As indicated by the relative positions of the nodes against the time-line in Figure 2.1, the occurrence of events follows a certain sequence in each time-step throughout the maintenance planning period: the system state $\mathbf{x}$ is assumed to stay the same within a time-step, and a transition of the system state can only potentially happen at the very beginning of a time-step in our model; the maintainers and operators observe the system state at the beginning of a given time-step (after any potential system state transition within the time-step), and simultaneously the two managerial parties make maintenance decision and production decision respectively, where the production decision also depends on the contracted demand for the time-step; following the maintenance and production decisions, the corresponding sales revenue, penalty cost and maintenance cost are spread across the rest of the given time-step; the selected maintenance action at a time-step may span across several following time-steps and thus directly influence the system state evolving process in the future time-steps.

The maintainers understand what kinds of decisions the operators are likely to take, and therefore from the perspective of the maintainers the operators follow a certain decision regime. Here we define *the operators' decision regime* as a machine utilisation policy which describes how the operators use the machine in response to a known demand and the machine state. Besides, the maintenance planning period is longer than the timescale of the production planning period, and therefore we choose to model the maintainers as the only decision maker in the system and build the machine utilisation behaviours of the operators into the maintainers' decision making model (we shall specify such a model in Section 2.2.4). Accordingly, the operators' decision node in Figure 2.1 (a) should be replaced by a random variable node as shown in Figure 2.2. Following such modelling



(a) In contracted period

(b) After contracted period

Figure 2.1: Maintenance planning problem structure

choice, the resulted maintainers' decision making model would not only capture the maintenance planning optimisation problem under the existing operators' decision regime, but also enable what-if analysis (i.e. updating the integrated operators' behaviours in the model) to assess how different operators' decision regimes would change the total costs and benefits. In other words, the resulted maintainers' decision making model has two-fold effects: facilitate the maintenance planning optimisation and improve the machine utilisation behaviours.

### 2.2.2 Mechanism of the operators' decision regime

The operators can change the speed of production by adjusting the rate of feeding raw materials into the system. For example, in the coal-fired power plants where coal is ground and then burnt and converted into electricity, the operators control the speed of coal feeding [7] in order to adjust the quantity of coal grinding; in the steel production plants

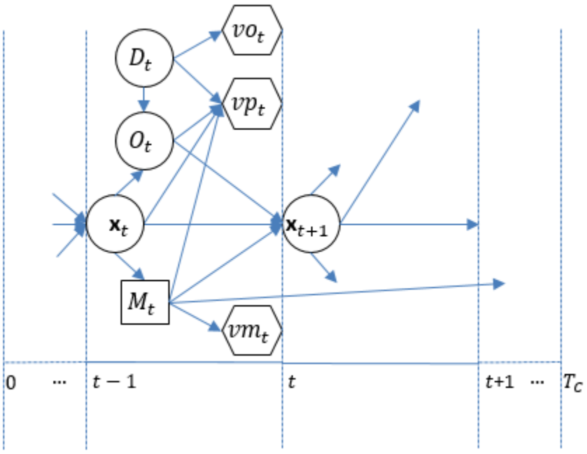


Figure 2.2: Maintenance problem structure: maintainers' view

where steel scraps are liquefied and refined and further processed into steel products, the operators control the speed of steel scraps feeding [15]. We refer to such feeding rate as the *work-rate* of the system. Naturally, the decision on work-rate should be based on the contracted demand and the operational performance of all the assets in the system.

In this study, we consider that the operators control the work-rate only at the system level. In reality, however, the operators may be able to further adjust the work-rate at the individual asset level. For example, in the coal-fired power plant which motivates this study, one production unit contains eight grinding mills and the operators are able to separately adjust the coal feeding speed of each individual mill. In practice, the operators may choose to only speed up the production of the mills with relative good states and leave the other mills with relative bad states to work at a normal speed, due to the concern that a higher than normal work-rate on the latter mills may force them to fail very quickly. In our future studies we may incorporate the possibility of individual asset work-rate adjustment.

We denote the work-rate of the system at time-step $t$ as $wr_t$. The operators' decision regime defines $wr$ on the contracted demand and the operational performance of all the assets in the production system. Hereafter in this thesis we assume that: (1) the maintainers understand the exact decision regime that the operators follow, such that $wr$

can be integrated into the mathematical model of the maintenance planning optimisation problem; (2) $wr$ can be categorised into ordered discrete levels. Note that because of the stochastic machine deterioration process, $wr$ evolves stochastically throughout the contracted period from the perspective of the maintainers.

### 2.2.3 Modelling the deterioration process

The machine maintenance literature contains multiple alternative frameworks to model the machine deterioration process, as discussed in Section 1.1.2.2. Here we select the Markov chain (also called discrete-time discrete-state Markov process in some studies) based on the three following assumptions: (1) the system state satisfies the Markovian property, meaning the future state of the system is dependent on its current state and is independent from previous system states; (2) the transition between any system states always consumes a fixed time length, and in the future studies we may adopt a framework which is more flexible in terms of the time setting (such as semi-Markov process or continuous-time Markov process) to model the machine deterioration process; (3) the performance and condition of every asset can each be categorised into ordered discrete levels based on the perceived overall degradation and overall operational effectiveness accordingly.

**Transition probabilities elicitation framework**

Regarding an asset in the system, its future state may depend on the current states of other assets in addition to its own current state: under certain operators' decision regimes, if some assets have relative bad performance the operators may speed up the work-rate of the whole system in order to meet the contracted demand but as a result all the assets in the system would degrade faster. Given the modelling choice of Markov chain, we propose eliciting the system state deterioration process as follows: elicit the transition probabilities between system states at each discrete level of work-rate.

It is however difficult to directly derive the transition probabilities between system states, because the system state is defined as a vector comprised of the condition and performance of every asset in the system, and similar challenges arise from problems involving complex systems in general [69]. In a mathematical language, suppose the system consists of $N$ assets which are indexed as asset $1, 2, ..., N$ respectively, and if we denote the condition and performance of the $n^{th}$ asset as $\mathbf{c}_n$ and $\mathbf{p}_n$ accordingly, then the system state is defined as $\mathbf{x} := (\mathbf{c}_1, \mathbf{p}_1, \mathbf{c}_2, \mathbf{p}_2, ..., \mathbf{c}_N, \mathbf{p}_N)$. It is naturally challenging to directly elicit the transition probabilities for $\mathbf{x}$ as it is a vector which consists of $2 * N$ variables.

Therefore we further propose decomposing the elicitation task to the level of the condition and performance transition of every single asset in the system: (1) The transition probabilities between condition levels of a single asset is elicited at each discrete level of the work-rate. (2) The transition probabilities between performance levels of a single asset, ideally, should be elicited at each discrete level of work-rate and each potential condition level, as both work-rate and condition can affect the transition rates between performance levels. As an approximation to such ideal elicitation, the transition probabilities between every two performance levels are elicited at each potential condition level given the normal work-rate level is applied. The dependence of the transition rates between performance levels on the work-rate is captured indirectly by recognising that the transition rates between performance levels are dependent on the asset's condition of which the transition directly depends on the work-rate.

Such elicitation framework is the generalisation result of a case-study specific elicitation framework from [7].

### 2.2.4 Modelling the maintenance planning optimisation problem

So far in this chapter we have discussed how to model the utilisation behaviours of the operators and the machine deterioration process. Based on such modelling work above, below we discuss how to model the maintenance planning optimisation problem: first we discuss the structure of the optimisation problem in Section 2.2.4.1; then in Section 2.2.4.2 we briefly explain a mathematical modelling framework which we shall use to model the optimisation problem of interest; finally in Section 2.2.4.3 we model the maintenance planning optimisation problem by applying such framework.

#### 2.2.4.1 Structure of the maintenance planning optimisation problem

The maintainers need to schedule a series of maintenance actions up to the planning-horizon, and the maintainer can choose from multiple potential maintenance alternatives. Each maintenance action addresses a particular set of condition-based and/or performance-based variables/measures. As part of the consequences of selecting a specific maintenance action, regarding the asset receiving the maintenance, the future asset state will be improved and potentially yield a longer lifetime and/or better future production performance (in terms of quantity and/or quality), while in the short-term the maintenance may cause some loss of production in addition to a fixed maintenance cost, and the potential loss of production may further lead to potential penalty cost or potential loss of sales, depending on whether the maintenance action is scheduled within or after the contracted period. Part of the uncertainties regarding the consequences of a given maintenance action roots in the stochastic nature of the demand after the contracted period; part of uncertainties are induced by the stochastic machine deterioration process. Additionally, the integrated operators' behaviours in the maintenance planning problem render the rate of machine deterioration evolving stochastically, which induces further uncertainties.

Therefore modelling the maintenance planning problem requires capturing the following: maintenance alternatives; potential consequences of choosing different maintenance alternatives; the uncertainties regarding which consequence will shows up. Additionally, in order to evaluate the relative superiority of different maintenance alternatives for arbitrary system states at any given time up to the planning-horizon, a utility function is needed in the mathematical model to reflect the total benefits/costs of different potential maintenance choices given the aforementioned uncertainties.

Given that in the conceptual framework (see Section 2.2.1) a discrete time-setting is assumed for both the machine deterioration process and the maintenance planning decision making and furthermore given Markov chain is selected to model the system state deterioration process (see Section 2.2.3), here we choose to model the maintenance planning optimisation problem based on *Markov decision processes (MDPs)* [82] in order to mathematically capture the aforementioned decision alternatives, consequences, uncertainties and utility function.

### 2.2.4.2 Markov decision processes

MDPs are a widely used mathematical framework for modelling decision making problems in which the state-evolution process of the underlying system is assumed to satisfy the Markov property. Below we specify the fundamental elements of MDPs [82], and in the next section we apply MDPs to modelling the maintenance planning optimisation problem of interest.

An MDP is defined by a finite set of discrete states $S$, the finite set of actions $A(\mathbf{x})$ permitted for state $\forall \mathbf{x} \epsilon S$, the reward function $R(\mathbf{x}, a, D, wr)$ which defines the immediate reward of applying action $\forall a \epsilon A(\mathbf{x})$ to state $\forall \mathbf{x} \epsilon S$ given the demand $D$ and the work-rate $wr$, and the probability of transition from state $\forall \mathbf{x} \epsilon S$ to state $\forall \mathbf{y} \epsilon S$ immediately after action $\forall a \epsilon A(\mathbf{x})$ where the probability is denoted as $P(\mathbf{y}|\mathbf{x}, a)$. In an MDP, the system state transition and decision making are assumed to occur at discrete time-steps.

In relation to the decision making problem, $A(\mathbf{x})$ models the decision alternatives given system state $\mathbf{x}$; $R(\mathbf{x}, a, D, wr)$ models part of the consequences of applying action $a$ to state $\mathbf{x}$ given the demand and work-rate; the other part of the consequences is the impact on the stochastic state transition which involves uncertainties (regarding which system state shows up next), and such impacts as well as the uncertainties of the transition are modelled by $P(\mathbf{y}|\mathbf{x}, a)$. Up to the planning-horizon, at each time-step, the stochastically evolving process generates an immediate reward which depends on the current state and selected action. A function that prescribes the appropriate action to be selected for every state given the the remaining time-length up to the planning-horizon is called a policy, denoted as $\pi$.

In terms of utility functions, specific choices include: total discounted rewards up to the planning-horizon, total rewards which is not discounted, and average reward per state transition. For detailed discussion readers are referred to [64]. Here we focus on the total discounted rewards, because we shall evaluate costs and benefits in terms of monetary values for our maintenance planning problem and it is sensible to discount such values at different time-steps into present values. The optimisation problem is therefore specified as identifying an optimal policy which maximises the expected total discounted rewards up to the planning-horizon starting from each state. In a mathematical language, the expected total discount rewards, $V$, for initial state $\mathbf{x}$, under policy $\pi$ is defined by Equation (2.1)

$$V^{\pi}(\mathbf{x}) := E[\sum_{t=1}^{T} \gamma^{t-1} R(\mathbf{x}_t, \pi(\mathbf{x}_t, t), D_t, wr_t)|\mathbf{x}_1 = \mathbf{x}], \tag{2.1}$$

where $T$ denotes the total number of time-steps up to the planning-horizon, $\mathbf{x}_t$ is the state at time-step $t$, $\pi(\mathbf{x}_t, t)$ is the action selected to be applied to state $\mathbf{x}_t$ at time-step $t$ under policy $\pi$, $D_t$ is the demand at time-step $t$, $wr_t$ is work-rate at time-step $t$, and $0 < \gamma < 1$ is the discount factor which measures future rewards in terms of current value. The optimal policy for the MDP, denoted as $\pi^*$, is thus defined as a policy which ensures $V^{\pi^*}(\mathbf{x}) \geqslant V^{\pi}(\mathbf{x})$ for $\forall\, \mathbf{x} \epsilon S$ and $\forall\, \pi$.

For an infinite planning-horizon MDP, where $T \rightarrow \infty$, the remaining time-length up to the planning-horizon is always infinite at any time-step. According to [12, 122], for any infinite planning-horizon MDP, at least one fixed policy which prescribes the appropriate action for each state independently from the time-step satisfies the definition of the optimal policy. For infinite planning-horizon MDPs, therefore, we focus on fixed policies. The expected total discount rewards, $V$, for initial state $\mathbf{x}$, under fixed policy $\pi$ is defined by Equation (2.2)

$$V^{\pi}(\mathbf{x}) := E[\sum_{t=1}^{\infty} \gamma^{t-1} R(\mathbf{x}_t, \pi(\mathbf{x}_t), D_t, wr_t) | \mathbf{x}_1 = \mathbf{x}]. \tag{2.2}$$

#### 2.2.4.3 Applying MDPs to our problem

Now we apply MDPs to modelling the maintenance planning optimisation problem of interest. In this study we focus on the infinite planning-horizon problem, that is $T \to \infty$, and the resulted mathematical model can be easily modified for finite planning-horizon problems.

The infinite maintenance planning period can be split into two parts: the first part is the contracted period which has a finite time-length $T_C$, and the demand in the first part has been contracted in advance and the operators may speed up production in the contracted period to avoid/mitigate potential penalty cost; the rest part of the maintenance planning period is infinite long, and we refer it as the non-contracted period in which the demand is stochastic and loss of sales may happen. Due to such two-part nature of the planning period, we need to accordingly modify the MDPs introduced above in order to model the maintenance planning optimisation problem of interest. We discuss the modifications below.

The reward function $R(\mathbf{x}, a, D, wr)$ is simplified as $R_{NC}(\mathbf{x}, a, D)$ for the non-contracted period, because the work-rate is assumed to stay at a normal level in the non-contracted period as a simplified modelling choice (discussed in Section 2.2.1).

Regarding the state transition probability $P(\mathbf{y}|\mathbf{x}, a)$, it further depends on the work-rate in the contracted period. We therefore modify the state transition probability as follows for the contracted period: $P_C(\mathbf{y}|\mathbf{x}, a, wr)$, and we denote the the state transition probability in the non-contracted period as $P_{NC}(\mathbf{y}|\mathbf{x}, a)$.

Additionally, instead of considering a uni-mode policy, we consider a maintenance policy $\pi$ that consists of two sub-policies each of which prescribes the appropriate maintenance actions to be selected throughout the contracted period or non-contracted period respectively, and we denote the sub-policy for contracted period as $\pi^C$ and denote the sub-policy for non-contracted period as $\pi^{NC}$. More specifically speaking, $\pi^C$ is a function that prescribes the appropriate action to be selected for every state given the the remaining time-length in the contracted period, and $\pi^{NC}$ prescribes the appropriate action for each state independently from the time-step.

Finally, we modify the value function. The expected total discount rewards, $V$, for initial state $\forall\,\mathbf{x}$, under fixed policy $\forall\,\pi := (\pi^C, \pi^{NC})$ is defined by Equation (2.3), which sums the expected discounted rewards of the contracted period and non-contracted period

$$V^{\pi}(\mathbf{x}) := E[\sum_{t=1}^{T_C} \gamma^{t-1} R_C(\mathbf{x}_t, \pi^C(\mathbf{x}_t, t), D_t, wr_t) + \sum_{t=T_C+1}^{\infty} \gamma^{t-1} R_{NC}(\mathbf{x}_t, \pi^{NC}(\mathbf{x}_t), D_t) | \mathbf{x}_1 = \mathbf{x}], \tag{2.3}$$

where $\pi^C(\mathbf{x}_t, t)$ is the action selected to be applied to state $\mathbf{x}_t$ at time-step $t$ under policy $\pi^C$ in the contracted period, $\pi^{NC}(\mathbf{x}_t, t)$ is the action selected to be applied to state $\mathbf{x}_t$ under fixed policy $\pi^{NC}$ in the non-contracted period, $D_t$ is the demand at time-step $t$ and $D_t$ is assumed as i.i.d. in the non-contracted period, $wr_t$ is work-rate at time-step $t$ and $wr_t$ is deterministically determined given $D_t$ and $\mathbf{x}_t$.

The optimisation problem is therefore specified as identifying an optimal policy $\pi^*$ which ensures $V^{\pi^*}(\mathbf{x}) \geqslant V^{\pi}(\mathbf{x})$ for $\forall\,\mathbf{x} \epsilon S$ and $\forall\,\pi$.

**Illustrative scenarios: contracted period and non-contracted period**

Here we provide some simplified problem scenarios to illustrate (1) how to specify the contracted period and non-contracted period numerically in the mathematical model when applying the PCM approach to real-life problems and (2) the importance of modelling based on different time scales (in other words, using both contracted period and non-contracted period in the model), in comparison with modelling only based on a single time scale.

Suppose a manufacturing company of interest would not be closed down in any foreseeable future and in addition currently the company has contracted the sales of production up to two months. The time length of the contracted period $T_C$ in the resulted mathematical model (referring to Equation 2.3) should be initialised as the total number of time-steps that the two contracted months contains (for example, assuming a time-step is measured as one week in the modelling work then $T_C$ should be set as 8 in the model) and the time length of the maintenance planning period $T$ should be initialised as infinite large (that is $\infty$ in mathematical notation) respectively. In a slightly different alternative scenario, suppose the company is scheduled to be closed down in five months, then the time length of the maintenance planning period $T$ should instead be initialised as the total number of time-steps the scheduled five months contains. Below we shall keep using the former case scenario to illustrate how to update the values for contracted period $T_C$ and maintenance planning period $T$ in the mathematical model as time rolls on in real-life problems.

Now suppose one week has passed since the start of the existing contract, and no new contracts have been made yet, then the contracted period $T_C$ in the mathematical model should be updated as 7 (assuming a time-step is measured as one week in the modelling work) and the maintenance planning period $T$ remains as infinite large (as long as the company would not be closed down in any foreseeable future). In a different (and more complex) alternative scenario for comparison, suppose one week has passed since the start of the existing contract, and a new contract (in addition to the existing one) with a time length of three months has just been agreed and has become effective (meaning the company must start supplying for the new contract immediately), then the contracted period $T_C$ in the mathematical model must be updated as 12, where the first 7 time-steps correspond to both the existing original contract and the newly agreed contract while the following 5 time-steps only correspond to the new contract; the maintenance planning period $T$ remains as infinite large in the mathematical model. In summary, both the contracted period $T_C$ and the maintenance planning period $T$ in the mathematical model are subject to adjustment as time rolls on in real-life in order to support decision making throughout the entire life-cycle of the company.

*What extra value can decision makers obtain by following the different-time-scale (contracted period versus non-contracted period)* natured *PCM approach, compared to following a simpler single-time-scale and myopic modelling approach to model the consequences of decision making only for the contracted period?* (Note in theory there exists a single-time-scale and *non*-myopic modelling approach as another alternative: directly capturing the potential work-rate adjustments and penalty costs beyond the contracted period in the modelling work, which means the operators' decision regime would directly extend to the non-contracted period in the model; hence the behaviours of the operators can also be directly modelled with a time-scale of the full life-cycle of the company, the same as the maintainers. But as discussed in Section 2.2.1, such a modelling choice would render the resulted mathematical model more complex rather than simpler, and therefore we shall not further discuss this modelling choice here.)

It is because for some real-life problems the models derived based on the myopic approach risks resulting in less profitable decision making which (1) recommends cheap (in terms of money and/ or time) maintenance choices to maintainers even though in practice more effective and expensive maintenance actions would actually yield better results (judging from the total costs/benefits of the full life-cycle) and (2) recommends high work-rate to operators to resolve potential production shortfalls even though paying the penalty cost could be more profitable. Below we provide more detailed explanation.

Based on the cost (in terms of money and/ or time) and effects, different maintenance choices can usually be approximately categorised into two types in practice: the ones that are relatively cheap to implement but bring limited improvements to the machine state and the benefits wear off relatively quickly; the ones that are expensive but can bring major improvements to machine state and the benefits last much longer. For example, an overhaul that restores the core part of machine system as good as new would be expensive, because an overhaul usually not only costs a relatively high price to implement (for instance the company may need to purchase expensive components) but also takes a long time to finish which may induce a non-trivial amount of production loss, but the benefits of having a machine system as good as new can last for a relatively long time during which the machine system can provide better production performance and are less likely to encounter unexpected failures. In comparison, a service that simply tunes the machine system costs a much lower price and is quick to implement, but the benefits (for instance temporarily enhanced production rate) of a tuned machine could be worn off quickly as well. When modelling the benefits of different maintenance choices, the myopic approach risks ignoring the full benefits of the expensive maintenance choices as their benefits may outlast the time-length of the mathematical model.

Regarding modelling the lasting consequences of speeding up the production work-rate, the myopic approach risks overlooking the negative impacts of an increased degradation rate on machine state evolving process beyond the limited time-length of the mathematical model. As a result, the myopic approach based models risk recommending speeding up the work-rate to resolve imminent production shortfalls even though in some real-life cases a better choice would be paying a penalty cost in the short-term and in turn saving a potentially large amount of maintenance cost and avoiding production interruption caused by unexpected machine failures in the long-term.

Later in Section 4.1.4, we shall conduct a simplified case study to numerically demonstrate that decision makers can encounter considerably high losses of expected value by following the myopic approach compared with adopting the PCM approach.

## 2.3 Comparison with relevant studies

The two crucial ideas that motivate the performance-centred maintenance (PCM) approach research line are (1) the necessities of separating degradation condition and operational performance as two entities and (2) balancing between machine utilisation and increased risk of machine failure via unifying the business functions of the operators and maintainers. In Section 1.2 above, we present a brief literature review on studies related to such two ideas and highlight their limitations from the perspective of the PCM approach. Here we shall further discuss the relationship between the PCM approach and the existing maintenance approaches from the reviewed studies.

From the perspective of modelling, the existing maintenance approaches are actually specific instances of the more sophisticated PCM approach. In more details, the existing maintenance approaches only fit for the type of problem scenarios where the condition and performance are perfectly correlated, and some approaches further assume that production has deterministic wearing-off effects on machines; in comparison, the PCM approach fits for different real-life problem scenarios where (1) the condition and performance evolving processes are either perfectly correlated, partially correlated or independent from each other and (2) the machine state is worn by production either in a deterministic pattern or in a stochastic pattern. Such flexible modelling capability of PCM approach derives from its sophisticated modelling framework: PCM approach (1) captures condition and performance as two separate entities in the modelling framework and (2) models the influence of decision making on machine state from a flexible stochastic perspective, and therefore it is simply a matter of adjusting the parameter values (more specifically speaking, adjusting the machine state transition matrices mostly) in the mathematical model to fit for different types of aforementioned problem scenarios.

Applying the less sophisticated maintenance approaches to complex problems they do not fit for risks deriving low quality decisions for operators and maintainers, since such maintenance approaches cannot effectively model some of the important problem features. The type of problems (referring to Section 2.1) investigated in this thesis is an example, and later in Section 4.1.4 we shall conduct simplified case studies to compare the performance (on modelling quality and decision making quality) between the less sophisticated maintenance approaches and the PCM approach.

## 2.4 Necessity for heuristics

In this chapter, we develop a new PCM approach to capture how the time-scale distinction between the operators and maintainers in their decision making impacts the gap between their value-perception of various operations activities. The PCM approach in general aims at balancing complex trade-offs between machine utilisation and increased risk of failure for better costs/benefits. The application of the PCM approach in specific case studies would however result in large size mathematical problems in the form of Markov decision processes (MDPs), as we shall see in the power plant case study in Chapter 4. It is impractical to solve the large size MDP problems by exact methods: for example the combination of dynamic programming (which is a type of exact computation method and we shall specify it in Chapter 3) and brute-force lookup tables method (which is an exact data-storage method), as exact methods would induce *impractically long* computational time and/or *impractically large* data-storage cost (we shall specify the terminology in this section). Heuristics are therefore required.

In Chapter 3, we shall evaluate and choose from some existing heuristic methods and in Chapter 4 we shall use the chosen heuristics as the input of our own design of heuristics for the power plant case study. More specifically speaking, in Chapter 3, we choose a heuristic computation method called *Q-learning*, which supposedly can approximately solve large size MDP problems in a *relatively short* computational time and significantly mitigate the data-storage cost associated with storing the state transition probabilities.

Additionally, in Chapter 3, we choose a data-storage heuristic called *polynomial function method* which aims at approximately storing the computational results produced when applying computation methods to solving large size MDPs, with *relatively small* data-storage cost and also *relatively small* sacrifice on data accuracy. Figure 2.3 summarises the functions of the two types of heuristics (i.e. computation heuristics and data-storage heuristics).

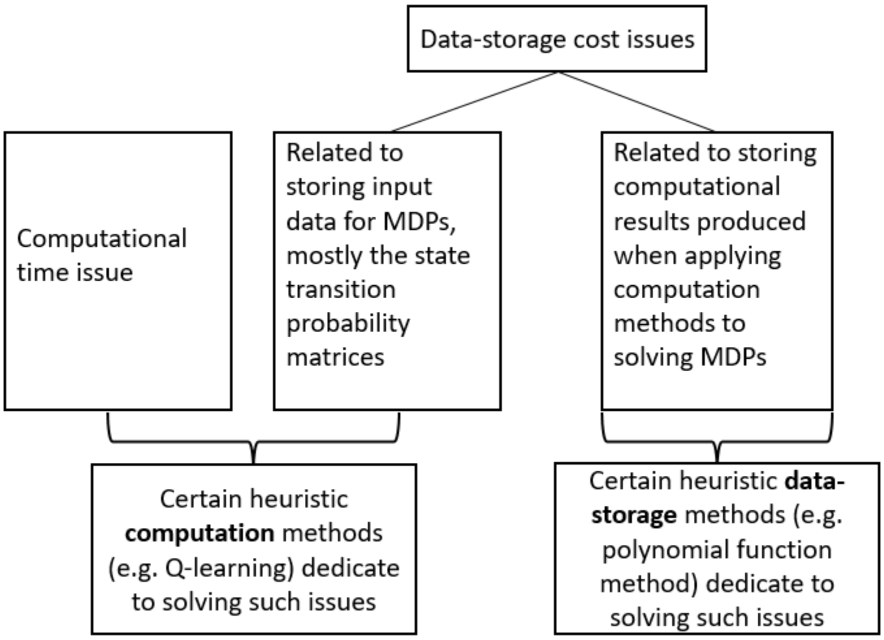

Data-storage cost issues
Computational time issue
Related to storing input data for MDPs, mostly the state transition probability matrices
Related to storing computational results produced when applying computation methods to solving MDPs
Certain heuristic **computation** methods (e.g. Q-learning) dedicate to solving such issues
Certain heuristic **data-storage** methods (e.g. polynomial function method) dedicate to solving such issues

Figure 2.3: Heuristics and the issues they dedicate to

Before presenting readers with technical discussions regarding the heuristics in the next chapter (Chapter 3), here we would like to discuss three measures which we shall refer to when evaluating the performance/effectiveness of different methods in solving the large size mathematical problems of interest; additionally, for each such measure we also specify the numerical benchmark and explain some relevant terms (that are already used in this section and shall be used repeatedly hereafter in this thesis).

Measure (1): computational time

The first measure is the total amount of computational time required in solving the mathematical problem of interest, given certain computation method(s) is implemented on a standard PC. We choose 24 hours as the numerical benchmark, based on multiple numerical cases reported in literature (such as [32, 56, 65, 66, 67, 70, 71, 72, 151]). Of course such number is purely a rule of thumb, and practitioners may need to adjust the benchmark in their own case studies. Additionally, when we refer to *a standard PC* in this thesis, we mean a PC with computation power and storage capacity close to the one that is used in this PhD project (we shall specify the physical measures of our PC in Chapter 5) as we assume that many practitioners/researchers rely on such kind of standard PCs in their work. Here we would like to clarity some relevant terminology:

- When we say the computational time of solving an MDP problem is *impractically long*, we mean the computational time takes longer than what practitioners accept and it usually happens if an exact computation method is applied to a large size MDP problem.

- When we say heuristic computation methods aim at solving an MDP problem in a *relatively short* computational time, we mean the computational time is both (1) acceptable to practitioners and (2) at least one magnitude less than applying exact computation methods.

Measure (2): data-storage cost

The second measure is the total data-storage cost induced in solving the mathematical problem of interest, given certain computation method(s) and data-storage method(s) are implemented. We set the numerical benchmark as 100% of maximum available memory space a standard PC provides. Here we would like to clarity some relevant terminology:

- When we say that the data-storage cost arising from solving the MDP problem of interest is *impractically large*, we mean the standard PC in use cannot store the data (including (1) the input data for the MDP problem, mostly the state transition probability matrices, and (2) the computational results that need to be stored when applying a computation method to solving the MDP problem) in the designated format (such as brute-force lookup tables); such data-storage issue usually happens if exact computation/data-storage methods are applied to large size MDP problems.

- Additionally, when we say heuristic data-storage methods aim at inducing *relatively small* data-storage cost in terms of storing the computational results, we mean the storage cost is (1) small enough such that the standard PC in use can handle it and (2) at least one magnitude less than applying the brute-force lookup tables method.
- When we say an MDP problem is "large size", we mean the MDP problem has a large set of states (in other words, a large state space) such that the computational time or the data-storage cost which arises from applying exact computation or data-storage methods to the MDP problem becomes impractically long/large. As a rule of thumb based on literature review and our own computation experience in this PhD project, usually a MDP problem with one million states is seen as large size when a standard PC is used. We would like to highlight that such threshold mostly holds for MDP problems with an infinite horizon; as for MDP problems with finite horizons, the threshold further decreases (and hence more challenging) following a reciprocal numerical relationship with the total number of time-steps in the planning period, because one extra dimension of cost (i.e. the time-step) is added into storing the computational results.

Measure (3): data accuracy level

The third measure is the the accuracy level of the stored computational results; note that computation heuristics and data-storage heuristics both sacrifice data accuracy in exchange for improvement on computational time and data-storage cost compared to exact methods. We set the numerical benchmark as 5% maximum data difference between the computational results derived by heuristics and the computational results derived by either exact methods or other benchmarking heuristics, based on aforementioned multiple numerical cases reported in literature. More specifically speaking, if exact methods are used as the benchmark methods, usually a small/medium-size version of the MDP problem would be tested; if other heuristics are used as the benchmark methods, either the full-size version or a medium-size version of the MDP problem would be tested. Such numerical benchmark is a rule of thumb, and practitioners/researchers may need to adjust the value in their own case studies. Here we would like to clarity some relevant terminology:

- When we refer to *relatively small* sacrifice on data accuracy when using heuristics, we mean the level of accuracy sacrifice is acceptable to the practitioners.

# Chapter 3

# Solving MDPs: computation methods and data-storage methods

The main purpose of this chapter is to specify two existing heuristics which we shall use as the input of our own design of heuristics in Chapter 4 for the power plant case study. One such existing heuristic is called *Q-learning*, and the other one is called *polynomial function method.* As discussed below, researchers and practitioners can choose from multiple methods to solve MDP problems, and therefore another aim of this chapter is providing some general ground rules which facilitate researchers and practitioners choosing appropriate methods for their own studies (such rules help us choose Q-learning and polynomial function method for our case study).

Below in Section 3.1 we provide a brief summary of the main computation methods to solve MDPs, and justify our choice of *Q-learning*; in Section 3.2 we shall take a detour and specify an exact computation method (called *value iteration*), in order to provide readers with crucial preliminary knowledge for understanding Q-learning (and we shall also use value iteration to solve part of the MDP problem for the power plant case study in Chapter 4); in Section 3.3 we specify Q-learning; in Section 3.4 we provide a brief summary of the main data-storage heuristics and justify why we choose *polynomial function method.*

## 3.1 Exact and heuristic computation methods (brief summary)

The exact computation methods for solving MDPs are categorised into two general classes: *dynamic programming (DP)* methods and linear programming. DP methods include *value iteration* [11, 12] and *policy iteration* [82]. We shall specify how the value iteration method works in Section 3.2. As for the other exact methods, we recommend textbooks including [64, 80, 118] to interested readers.

Some heuristic computation methods combine DP methods with auxiliary heuristic techniques. Depending on the auxiliary heuristic techniques, such heuristic computation methods can be approximately classified into two categories: (1) the ones that reduce the size of the state space for MDPs and then apply DP methods to the reduced-size MDPs; (2) the ones that modify the default sequence of computation process embedded in DP methods in hope for faster convergence to near-optimal solutions. More specifically speaking, type (1) heuristics adopt the techniques of states aggregation which approximately judge the homogeneity (usually based on the background knowledge of the case study) of the states and then aggregate homogeneous states into mega states and redefine the MDP problem on the aggregated level [64]. Some recent example studies include [83, 114, 116, 136, 138]. Type (1) heuristics are however usually case specific and therefore difficult to be applied to other cases. Type (2) heuristics adopt the techniques of topological graphing which highlight the most likely state transitions during the system state evolving process and utilises such graphical structures to modify the default sequence of computation process in DP methods. Example studies include [34, 35, 36, 37, 38]. Type (2) heuristics however can impose impractically large computational and data-storage costs to construct and store such graphical structures.

Another type of heuristic computation methods called *reinforcement learning (RL)*

[133] encompasses so far arguably the most cutting-edge research in this area. Specific RL methods include Q-learning [148], SARSA [123], Q-P-learning [64], CAP-I [13] and actor-critic methods [133] and so on. The essence of (most if not all) RL methods is combining two techniques: (1) simulating the system state evolving process for the MDP problem and (2) incremental value-updating based on the so called *temporal difference* [132]. The technical difference between specific RL methods consists in how such two techniques are implemented. We shall specify the two techniques in the context of Q-learning in Section 3.3, and here we avoid any discussion regarding such two techniques in a more general context in case of distracting readers from the main purpose of this thesis. For readers who are however interested, we refer to [64, 69, 118, 133].

**Criteria for selecting computation methods**

Given such multiple computation methods as alternatives, here we propose three criteria to help researchers/practitioners make proper choice for their cases: (1) the methodology of the method does not require any impractical exploitation on problem specific knowledge, (2) the method has solid mathematical proof to guarantee the derived solutions converge to optimum under certain conditions and (3) abundant numerical tests exist in publications to empirically suggest such method can indeed approximately solve large size MDP problems in a relative short time and find relatively accurate solutions. Such criteria reduce the alternatives to Q-learning and SARSA for our case study; it is possible other case studies may possess unique problem structures which lead to preference for other heuristics (for instance state aggregation heuristic).

We would like to clarify that such criteria are proposed as a rule of thumb for researchers and practitioners to quickly choose a computation heuristic to solve the large size MDP problems in their own case studies, rather than as a set of rules to discriminate one heuristic against another in *theoretical* research: due to a lack of theoretical research, it is unclear whether the computation of some heuristic methods is guaranteed to converge in theory, and hence such methods would so far be less favoured by practitioners in many specific case studies following such three criteria; however, further theoretical research may prove some of such methods can converge towards optimal results faster than proved heuristics (in other words, capable of providing better empirical results in terms of computational time).

We choose Q-learning for the power plant case study (which we shall discuss in Chapter 4). As discussed in Section 2.3, Q-learning corresponds to (1) the issue of impractically long computational time and (2) the issue of impractically large data-storage cost associated with the input data for MDPs (more specifically speaking, the state transition probability matrices) which arise from solving large size MDPs with exact methods. First we shall lay the groundwork and specify the exact computation method *value iteration (VI)* in Section 3.2, and then we shall specify Q-learning in Section 3.3 and discuss how it handles the two issues aforementioned.

## 3.2 Value iteration

Value iteration (VI) relies on solving the so called Bellman optimality equation to obtain optimal policies for MDPs (the concept of optimal policies for MDPs is explained in Section 2.2.4.2). Such equation is a recursive relation defined below for finite and infinite planning-horizon MDPs separately.

### 3.2.1 VI for finite-horizon MDPs

For an MDP with a finite planning-horizon $T$, the optimal value of state $\forall\, \mathbf{x} \epsilon S$ at time-step $\forall\, t$, $V_t^*(\mathbf{x})$, is defined by Equation (3.1)

$$V_t^*(\mathbf{x}) = \max_{a \epsilon A(\mathbf{x})} \{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V_{t+1}^*(\mathbf{y})\}, \tag{3.1}$$

which seeks the maximum of the immediate reward from applying an action to the current state plus the expected optimal value over the remaining planning period. Equation (3.1) is the so called Bellman optimality equation for finite planning-horizon MDPs and the $V^*$ terms are the unknowns in the equation. As shown in [118], $V^*_{t=1}(\mathbf{x})$ from Equation (3.1) is equal to the maximal objective function value $V^{\pi^*}(\mathbf{x})$ from Equation (2.1) for $\forall\, \mathbf{x}$. Solving Equation (3.1) and obtain the optimal value for $\forall\, (\mathbf{x}, t)$ therefore enables identifying the optimal policy $\pi^*$ for finite planning-horizon MDPs.

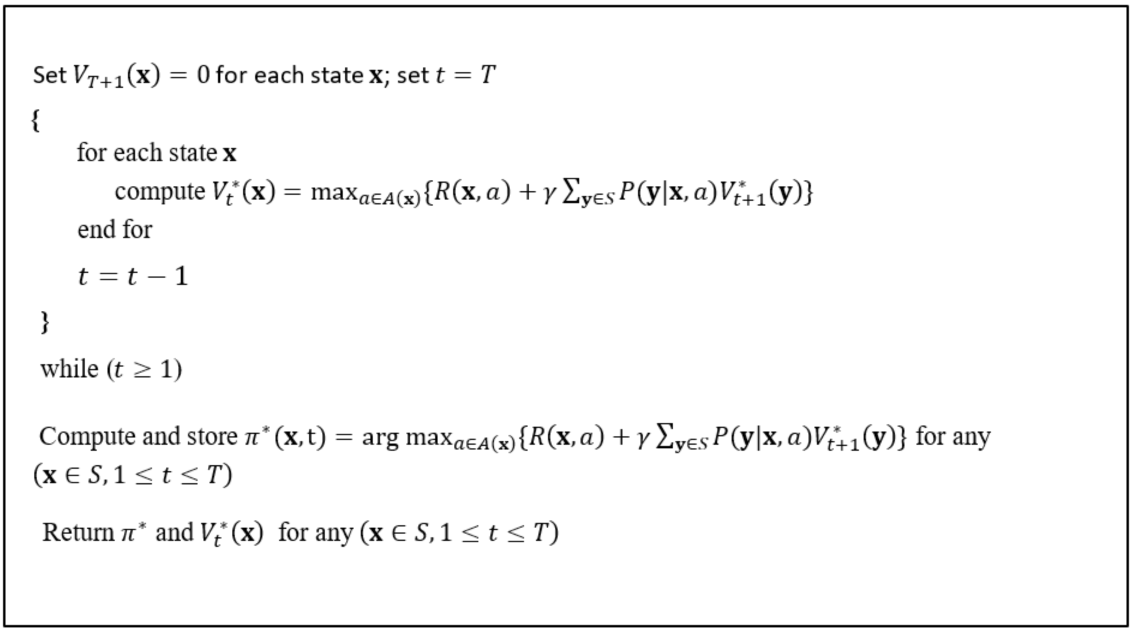
For finite planning-horizon MDPs, the value iteration method starts from the last time-step of the mathematical model and works backwards in time through the entire planning period. At each time-step, VI visits all the states one by one and calculates the exact optimal value for each state at that time-step by using Equation (3.1), and these optimal values are the input for calculations at the time-step to be visited next. Such process continues until the first time-step is visited. The optimal policy is identified from these optimal values at each time-step, more specifically that is $\pi^*(\mathbf{x}, t) := \arg\max_{a \epsilon A(\mathbf{x})} \{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V^*_{t+1}(\mathbf{y})\}$ for $\forall\, \mathbf{x} \epsilon S$ and $\forall\, t \leq T$. Pseudo-code of VI for finite planning-horizon MDPs is given in Figure 3.1.

### 3.2.2 VI for infinite-horizon MDPs

For an infinite planning-horizon MDP, since the remaining time-length up to the planning-horizon at any time-step is always infinite, it is meaningless to specify the time-step. Equation (3.1) therefore is updated as Equation (3.2) accordingly: for state $\forall\, \mathbf{x} \epsilon S$

$$V^*(\mathbf{x}) = \max_{a \epsilon A(\mathbf{x})} \{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V^*(\mathbf{y})\}. \tag{3.2}$$

It is shown by [118] that $V^*(\mathbf{x})$ from Equation (3.2) is equal to the maximal objective function value $V^{\pi^*}(\mathbf{x})$ from Equation (2.2) for $\forall\, \mathbf{x}$. Equation (3.2) is the so called Bellman optimality equation for infinite planning-horizon MDPs and the $V^*$ terms are the



```
Set $V_{T+1}(\mathbf{x}) = 0$ for each state $\mathbf{x}$; set $t = T$
{
    for each state $\mathbf{x}$
        compute $V_t^*(\mathbf{x}) = \max_{a \in A(\mathbf{x})}\{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \in S} P(\mathbf{y}|\mathbf{x}, a) V_{t+1}^*(\mathbf{y})\}$
    end for
    $t = t - 1$
}
while $(t \geq 1)$

Compute and store $\pi^*(\mathbf{x}, \mathrm{t}) = \arg\max_{a \in A(\mathbf{x})}\{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \in S} P(\mathbf{y}|\mathbf{x}, a) V_{t+1}^*(\mathbf{y})\}$ for any
$(\mathbf{x} \in S, 1 \leq t \leq T)$

Return $\pi^*$ and $V_t^*(\mathbf{x})$ for any $(\mathbf{x} \in S, 1 \leq t \leq T)$
```

Figure 3.1: VI finite planning-horizon [118]

unknowns in the equation. Solving Equation (3.2) and obtain the optimal value for $\forall\, \mathbf{x}$ therefore enables identifying the optimal policy $\pi^*$ for infinite planning-horizon MDPs.

For infinite planning-horizon MDPs, VI initiates the *estimate* of the optimal value, $V^*(\mathbf{x})$, as an arbitrary value (usually zero or a small positive value) for every state $\mathbf{x} \epsilon S$, and then iteratively updates these estimates until a pre-set termination criterion is met. In each iteration, VI visits all the states one by one and updates the estimate for each state based on a transformation derived from Equation (3.2), and the updated estimates are the input for the computations in the next iteration. The derived transformation is as follows: Let $V_k(\mathbf{x})$ denote the estimate of optimal value $V^*(\mathbf{x})$ for state $\mathbf{x}$ updated in the $k^{th}$ iteration of VI, then $V_{k+1}(\mathbf{x})$ in the next iteration is calculated by Equation (3.3). The ultimate policy is derived from the estimates updated in the last iteration of VI.

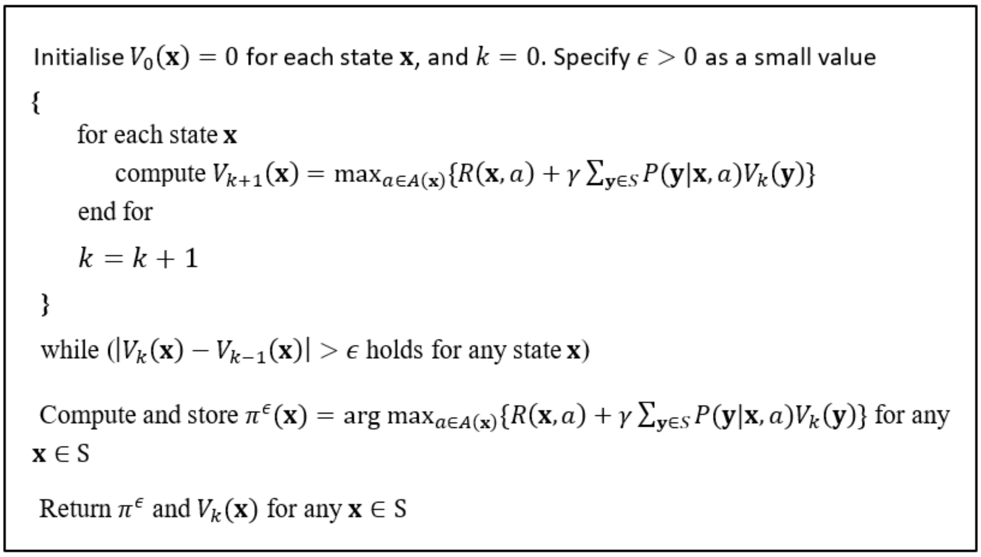
$$V_{k+1}(\mathbf{x}) = \max_{a \epsilon A(\mathbf{x})} \{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V_k(\mathbf{y})\}. \tag{3.3}$$

The estimate of the optimal value in VI converges to the exact optimal value for each state, that is converging to $V^*(\mathbf{x})$ for $\forall\, \mathbf{x} \epsilon S$, given infinite number of iterations. The ultimate policy is therefore the optimal policy. But in practice no one can afford infinite computational time, and a termination criterion should thus be set to ensure a limited number of iterations and a certain level of computation accuracy.

An arbitrarily pre-set positive value is used as the termination criterion, and it is often denoted as $\epsilon$ in literature. As shown in multiple publications (for example [64, 118]), as long as $\epsilon > 0$ and the discount factor in an MDP is $0 < \gamma < 1$, after enough but a finite number of iterations, the optimal value estimate for each state from the last two iterations of VI would be close enough to ensure $\max_{\mathbf{x} \epsilon S} |V_K(\mathbf{x}) - V_{K-1}(\mathbf{x})| < \epsilon$ for $\forall\, \mathbf{x} \epsilon S$, and the policy derived by VI is close to the optimal policy which guarantees $\max_{\mathbf{x} \epsilon S} |V^{\pi^\epsilon}(\mathbf{x}) - V^{\pi^*}(\mathbf{x})| < 2\gamma\epsilon/(1-\gamma)$ for $\forall\, \mathbf{x} \epsilon S$, where $K$ is the total number of iterations that VI takes, $\pi^\epsilon$ is the policy that VI derives based on the estimates updated in the final iteration: more specifically speaking, $\pi^\epsilon(\mathbf{x}) := \arg\max_{a \epsilon A(\mathbf{x})} \{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V_K(\mathbf{y})\}$ for $\forall\, \mathbf{x} \epsilon S$. In summary, with enough but finite iterations, VI returns the decision maker with a near optimal policy which guarantees for each state that the gap between the value actually expected to be obtained (by following such near optimal policy) and the ideal optimal value (from following the exact optimal policy) is bounded by a certain threshold. Pseudo-code of VI for infinite planning-horizon MDPs is given in Figure 3.2.

### 3.2.3 Application issues of value iteration

Applying value iteration (VI) to large size MDPs is impractical. The optimal values (or their estimates) in VI are updated by brute-forces: in each iteration, VI visit all the states one by one and perform value calculation for each state based on Equation (3.1) or (3.3). For large size MDPs, however, the calculation for just one iteration can be very time consuming. Additionally, DP methods requires the transition probability matrices to be stored (each transition matrix consists of the transition probabilities from all states to all states given a certain action), but the memory burden of storing them can be impractically large for large size MDPs. Below we introduce Q-learning and discuss how



```
Initialise $V_0(\mathbf{x}) = 0$ for each state $\mathbf{x}$, and $k = 0$. Specify $\epsilon > 0$ as a small value
{
    for each state $\mathbf{x}$
        compute $V_{k+1}(\mathbf{x}) = \max_{a \in A(\mathbf{x})}\{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \in S} P(\mathbf{y}|\mathbf{x}, a) V_k(\mathbf{y})\}$
    end for
    $k = k + 1$
}
while ($|V_k(\mathbf{x}) - V_{k-1}(\mathbf{x})| > \epsilon$ holds for any state $\mathbf{x}$)

Compute and store $\pi^\epsilon(\mathbf{x}) = \arg\max_{a \in A(\mathbf{x})}\{R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \in S} P(\mathbf{y}|\mathbf{x}, a) V_k(\mathbf{y})\}$ for any
$\mathbf{x} \in \mathrm{S}$

Return $\pi^\epsilon$ and $V_k(\mathbf{x})$ for any $\mathbf{x} \in \mathrm{S}$
```

Figure 3.2: VI infinite planning-horizon [118]

Q-learning mitigate such two issues.

## 3.3 Q-learning

Q-learning derives solutions for MDP problems by *approximately* solves the Bellman optimality equation. As empirically suggested by multiple numerical studies in publication (e.g. [56, 65, 66, 67, 68, 70, 71, 72, 119, 131, 151]), usually Q-learning can derive relatively accurate computational results in a relatively short computational time for large size MDPs. Additionally, as proved by [149], the computational results are guaranteed to converge *as* exact results (hence deriving optimal policies) under certain conditions (we shall specify such conditions in Section 3.3.3).

We shall specify how Q-learning derives solutions for finite and infinite planning-horizon MDPs respectively in Section 3.3.1 and Section 3.3.2. In Section 3.3.3, we shall discuss why in many reported numerical studies Q-learning can approximately solve large size MDPs with relatively short computational time and relatively small data-storage cost related to transition probability matrices; meanwhile we shall also briefly discuss optimality properties of Q-learning.

### 3.3.1 Q-learning for finite-horizon MDPs

Q-learning for finite-horizon MDPs applies to the MDP problem which has a known and fixed initial state and a finite planning period. Such method aims at deriving near-optimal actions for the initial state and each state that the system can potentially transit into at each following time-step towards the planning-horizon. In this section, first we shall explain the methodology of Q-learning for finite-horizon MDPs and then we specify the technical details of such method. Hereafter in this section, by Q-learning we mean Q-learning for finite-horizon MDPs, unless stated otherwise.

Methodology of Q-learning for finite-horizon MDPs

Q-learning approximately solves the Bellman optimality equation in finite-horizon MDPs (Equation (3.2.1)) by (1) decomposing the Bellman optimality equation as below and then (2) approximating a type of intermediate values in such decomposed Bellman optimality equations. Finally, actions are selected based on comparing such intermediate value estimates between different actions.

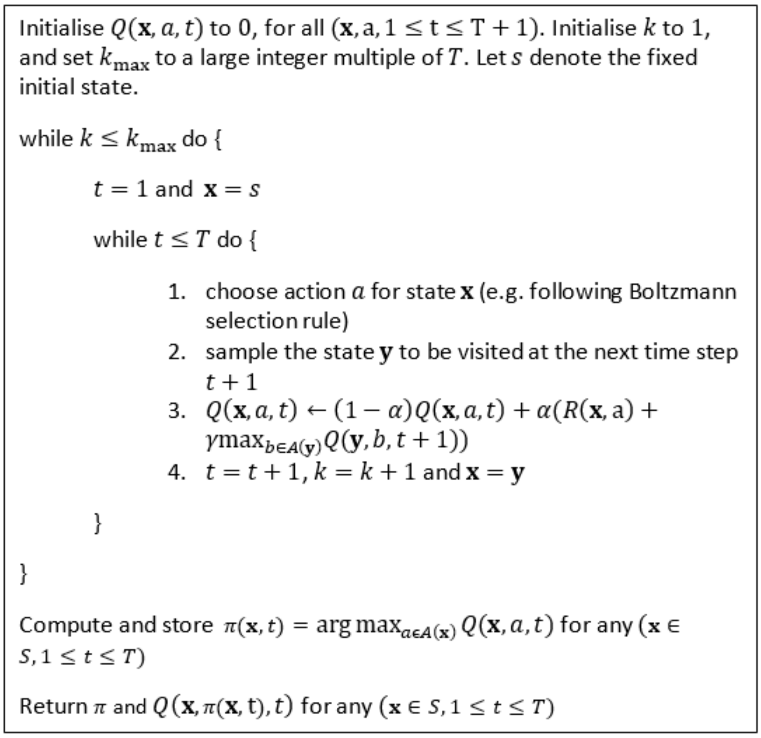
**Decomposed Bellman equations for finite planning-horizon MDPs**: for state $\mathbf{x}\epsilon S$, time-step $t \leq T$

$$V_t^*(\mathbf{x}) = \max_{a\epsilon A(\mathbf{x})} \{V_t^a(\mathbf{x})\}, \tag{3.4}$$

where

$$V_t^a(\mathbf{x}) = R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y}\epsilon S} P(\mathbf{y}|\mathbf{x}, a) V_{t+1}^*(\mathbf{y}). \tag{3.5}$$

The value denoted as $V_t^a(\mathbf{x})$ in the *decomposed* Bellman optimality Equations (3.4)-(3.5) is the intermediate value of interest, and it is called the expected value of state $\mathbf{x}$ under action $a$ at time-step $t$. Q-learning approximates such expected values for each state-action-time-step combination in the MDP problem, by incrementally updating the estimates of such expected values based on simulation (we shall explain the simulation-updating process below). Once such value estimates converge, Q-learning stops the simulation-updating process and constructs a policy by selecting the action with the highest estimate value for each state-time-step pair in the MDP problem. Pseudo-code of Q-learning for finite planning-horizon MDPs is given in Figure 3.3.



```
Initialise $Q(\mathbf{x}, a, t)$ to 0, for all $(\mathbf{x}, \mathrm{a}, 1 \le \mathrm{t} \le \mathrm{T} + 1)$. Initialise $k$ to 1,
and set $k_{\max}$ to a large integer multiple of $T$. Let $s$ denote the fixed
initial state.

while $k \le k_{\max}$ do {

        $t = 1$ and $\mathbf{x} = s$

        while $t \le T$ do {

                1. choose action $a$ for state $\mathbf{x}$ (e.g. following Boltzmann
                   selection rule)
                2. sample the state $\mathbf{y}$ to be visited at the next time step
                   $t + 1$
                3. $Q(\mathbf{x}, a, t) \leftarrow (1 - \alpha)Q(\mathbf{x}, a, t) + \alpha(R(\mathbf{x}, \mathrm{a}) +$
                   $\gamma \max_{b \in A(\mathbf{y})} Q(\mathbf{y}, b, t + 1))$
                4. $t = t + 1, k = k + 1$ and $\mathbf{x} = \mathbf{y}$

        }

}

Compute and store $\pi(\mathbf{x}, t) = \arg\max_{a \in A(\mathbf{x})} Q(\mathbf{x}, a, t)$ for any $(\mathbf{x} \in$
$S, 1 \le t \le T)$

Return $\pi$ and $Q(\mathbf{x}, \pi(\mathbf{x}, \mathrm{t}), t)$ for any $(\mathbf{x} \in S, 1 \le t \le T)$
```

Figure 3.3: Q-learning finite planning-horizon [70]

More specifically speaking, Q-learning initialises the value estimate as a random small value (or zero) for each state-action-time-step combination in the MDP problem which has a unique/fixed initial state, and then Q-learning starts the simulation-updating process from the the first time-step of the MDP problem. The simulation and the value estimates updating are intertwined in Q-learning. For an easier understanding, we shall first explain the simulation process and then explain how the estimates updating process is embedded in the simulation process. Note here we aim at providing a relative high-level view and we shall specify the technical details later.

The simulation process starts from the first time-step of the MDP problem and proceeds through each time-step in the MDP problem until it reaches the planning-horizon and then the simulation process restarts from the first time-step of the MDP problem. Such simulation process repeats for an arbitrarily pre-set large number. At each time-step in each iteration during the simulation process, only one state-action pair is randomly sampled (except that for the first time-step of the MDP problem the unique initial state is deterministically chosen in the simulation): later we shall specify the technical details regarding how the state and the action are sampled at each time-step.

Q-learning further embeds value estimates updating in such simulation process: at each time-step during the simulation process, once a state-action pair is sampled, Q-learning pauses the simulation process and updates the value estimate for such state-action-time-step combination and then continues with the simulation process. Each update adjusts the existing value estimate of the state-action-time-step based on an approximation of Equation (3.5), and later we shall specify the technical details of such an approximation.

In summary, the simulation-updating process starts from the initial time-step and progresses forwards to the last time-step and then continues from the initial time-step again until a pre-set number of iterations is met. In practice, the pre-set number should be sufficiently large to ensure the convergence of the value estimates. The value estimates updated in each iteration of simulation is the input of computation in the next iteration of simulation. Once the simulation-updating process is finished, Q-learning constructs a policy by selecting the action with the highest estimate value for each state-time-step pair in the MDP problem.

Below we specify the technical details of the method.

Transition matrices and state selection

Q-learning assumes (1) the system state is defined by the values of a set of individual random variables that govern the system dynamics and (2) in a complex system the transition matrices (see definition in Section 3.2.3) of the system states are difficult to derive in practice and the associated storage cost is impractically large whereas the distribution of each such individual random variable is relatively easy to derive in practice and the associated storage cost is relatively small [64]. Therefore Q-learning requires the distributions of such individual random variables are derived (rather than the transition matrices of the system states). Our elicitation framework (see Section 2.2.3) regarding state transition probabilities of the generic manufacturing system is an illustrative example: in such framework we decompose the elicitation work from the system level to the level of the condition and performance of every single asset in the system.

In the simulation process of Q-learning, the sampling of system state at a given time-step is based on (1) such distributions and (2) the state-action pair sampled by Q-learning at the previous time-step: the system state at the previous time-step specifies the value of each governing random variable at the previous time-step, and the governing variables' values at the current time-step are randomly sampled from the corresponding distributions given the previous values of such variables and the action sampled at the previous time-step. The new values of all the governing variables defines the system state at the given/current time-step.

Action selection

In the simulation process of Q-learning, the sampling of action at a given time-step is based on the most up-to-date value estimates: given a state sampled for a time-step, the actions associated with higher estimate values are more likely to be selected. General selection rules include $\epsilon$-greedy policy and softmax policies (such as Boltzmann selection rule which adjusts the probability of action selection as follows: $P(a|\mathbf{x},t) = e^{Q(\mathbf{x},a,t)/M} / \Sigma_{b \epsilon A(\mathbf{x})} e^{Q(\mathbf{x},a,t)/M}$, where $e$ is the base of the natural logarithm and $M$ is an input parameter which as a rule of thumb should be set as both large enough to ensure close to equal selection probabilities between all actions in the early stage of the simulation process of Q-learning and small enough to ensure the action selection converges towards the actions that have relatively high estimate values in the late stage of the simulation process) [64]. A drawback of these rules is they require parameter(s) tuning for each specific MDP problem and such tuning work can be potentially time-consuming. Alternatively, pure exploration policy under which all actions would equally likely be selected can also be used, and such policy does not require parameter(s) tuning but such policy may render the estimate values convergence slower in the application of Q-learning.

Approximation of Equation (3.5)

In Q-learning, the exact expected value $V_t^a(\mathbf{x})$ in Equation (3.5) is not computed. Instead, its corresponding estimate, denoted as $Q(\mathbf{x}, a, t)$, is incrementally updated as follows: suppose the currently selected state-action pair at time step $t$ is $(\mathbf{x}, a)$ and the next state to be visited by Q-learning at time step $t + 1$ is randomly sampled to be $\mathbf{y}$, then

$$Q(\mathbf{x}, a, t) \leftarrow (1 - \alpha)Q(\mathbf{x}, a, t) + \alpha(R(\mathbf{x}, a) + \gamma \max_{b \epsilon A(\mathbf{y})} Q(\mathbf{y}, b, t + 1)) \tag{3.6}$$

where $0 < \alpha < 1$ denotes a step-size parameter (we shall further discuss how to control such parameter below), $b$ denotes an arbitrary action that is permitted from state $\mathbf{y}$, $A(\mathbf{y})$ denotes the set of actions permitted from state $\mathbf{y}$, and $\leftarrow$ means that the value of the expression on the right-hand side is calculated and then used to replace the value of the variable on the left-hand side; $Q(\mathbf{x}, a, t)$ on the right-hand side of the equation represents the value estimate before update whereas $Q(\mathbf{x}, a, t)$ on the left-hand side represents the value estimate after update; all the other notations have the same meaning as in Section 3.2.1. The value estimate update consists of a proportion of the current value estimate in addition to an approximation of Equation (3.5): the approximation utilises $\max_{b \epsilon A(\mathbf{y})} Q(\mathbf{y}, b, t+1)$ as a proxy of $V_{t+1}^*(\mathbf{y})$ in Equation (3.5), that is using the maximum value estimate of the randomly sampled state to be visited next in simulation as a proxy for the actual optimal value over the remaining planning period of the MDP model.

Note by simply transforming the right-hand side of Expression 3.6 we can obtain the following: $Q(\mathbf{x}, a, t) + \alpha(R(\mathbf{x}, a) + \gamma \max\limits_{b \epsilon A(\mathbf{y})} Q(\mathbf{y}, b, t+1) - Q(\mathbf{x}, a, t))$, where $R(\mathbf{x}, a) + \gamma \max\limits_{b \epsilon A(\mathbf{y})} Q(\mathbf{y}, b, t+1) - Q(\mathbf{x}, a, t)$ is the so called temporal difference in literature (mentioned in Section 3.1) in the context of Q-learning .

Control of the step-size parameter

Regarding the control of the step-size parameter $\alpha$, Gosavi [64] recommends controlling rules such as $\frac{M}{N+k}$, $\frac{\ln(k+1)}{k+1}$ and the Darken-Chang-Moody rule [39] where $M$ and $N$ are large positive constants, $k$ denotes the $k^{th}$ sampling in the application of Q-learning and $\ln(k)$ represents the natural logarithm of $k$. Note all such controlling rules ensure $\alpha$ gradually decay in the application of $Q$-learning.

### 3.3.2 Q-learning for infinite-horizon MDPs

Q-learning for infinite-horizon MDPs applies to the MDP problem which has an infinite planning period. Such method aims at deriving near-optimal actions for every state of the MDP problem. In this section, first we shall explain the methodology of Q-learning for infinite-horizon MDPs and then we specify the technical details of such method. Hereafter in this section, by Q-learning we mean Q-learning for infinite-horizon MDPs, unless stated otherwise.

Methodology of Q-learning for infinite-horizon MDPs

Q-learning approximately solves the Bellman optimality equation in infinite-horizon MDPs by (1) decomposing the Bellman optimality equation as below and then (2) approximating a type of intermediate values in such decomposed Bellman optimality equations.

**Decomposed Bellman equations for infinite planning-horizon MDPs**: for state $\forall\, \mathbf{x} \epsilon S$

$$V^*(\mathbf{x}) = \max_{a \epsilon A(\mathbf{x})} \{V^a(\mathbf{x})\}, \tag{3.7}$$

where

$$V^a(\mathbf{x}) = R(\mathbf{x}, a) + \gamma \sum_{\mathbf{y} \epsilon S} P(\mathbf{y}|\mathbf{x}, a) V^*(\mathbf{y}). \tag{3.8}$$

The value denoted as $V^a(\mathbf{x})$ in the *decomposed* Bellman optimality Equations (3.7)-(3.8) is the intermediate value of interest, and it is called the expected value of state $\mathbf{x}$ under action $a$. Q-learning approximates such expected values for each state-action pair in the MDP problem, by incrementally updating the estimates of such expected values based on simulation (we shall explain the simulation-updating process below). Once such value estimates converge, Q-learning stops the simulation-updating process and constructs a policy by selecting the action with the highest estimate value for each state in the MDP problem. Pseudo-code of Q-learning for infinite planning-horizon MDPs is given in Figure 3.4.

More specifically speaking, Q-learning initialises the value estimate as a random small value (or zero) for each state-action pair in the MDP problem, and then Q-learning starts the simulation-updating process from an arbitrary initial state in the MDP problem. The simulation and the value estimates updating are intertwined in Q-learning. Similar to Section 3.3.1, for an easier understanding, we shall first explain the simulation process and then explain how the estimates updating process is embedded in the simulation process. Note here we aim at providing a relative high-level view and we shall specify the technical details later.

The simulation process proceeds towards the infinite planning-horizon of the MDP problem, and in each time-step during the simulation process only one state-action pair is randomly sampled. Q-learning terminates the simulation process after simulating a pre-set number of time-steps for the MDP problem. Q-learning further embeds value estimates updating in such simulation process: at each time-step during the simulation process, once a state-action pair is sampled, Q-learning pauses the simulation process and updates the value estimate for such state-action pair and then continues with the simulation process. Each update adjusts the existing value estimate of the state-action pair based on an approximation of Equation (3.8).

In summary, the simulation-updating process starts from an arbitrary initial state and progresses forwards towards the infinite planning-horizon of the MDP problem until a pre-set number of time-steps is met (note the number should be sufficiently large to ensure the convergence of the value estimates). The value estimates updated in each time-step is the input of computation in the next time-step of the simulation. Once the simulation-updating process is finished, Q-learning constructs a policy by selecting the action with the highest estimate value for each state in the MDP problem.

Regarding how the state-action pairs are sampled at each time-step during the simulation process, the corresponding technical details are identical to Section 3.3.1, and the only technical difference is the estimate values of state-action pairs are used here rather than the estimate values of state-action-time-step combinations. Below we specify the technical details regarding how the value estimates are updated.

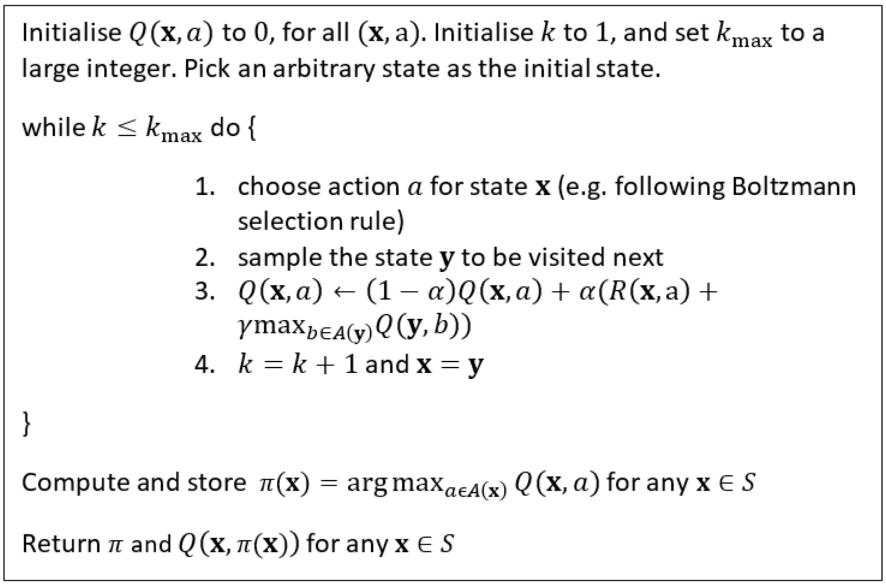
Approximation of Equation (3.8)

For infinite planning-horizon MDPs, Q-learning incrementally update the approximation of the expected value $V^a(\mathbf{x})$ defined in Equation (3.8), $Q(\mathbf{x}, a)$, as follows

$$Q(\mathbf{x}, a) \leftarrow (1 - \alpha)Q(\mathbf{x}, a) + \alpha(R(\mathbf{x}, a) + \gamma \max_{b \epsilon A(\mathbf{y})} Q(\mathbf{y}, b)) \tag{3.9}$$

where the notations have similar meaning as their counterparts in Section 3.3.1 (note the controlling rules discussed in Section 3.3.1 regarding the step-size parameter $\alpha$ also apply here), and the difference is here the time-step is not specified because the interest is in value estimation for each state-action pair, over an infinite number of remaining time-steps; additionally, $Q(\mathbf{x}, a)$ on the right-hand side of the equation represents the value estimate before update whereas $Q(\mathbf{x}, a)$ on the left-hand side represents the value estimate after update. The value estimate update consists of a proportion of the current estimation value in addition to an approximation of Equation (3.8): the approximation utilises $\max\limits_{b\epsilon A(\mathbf{y})} Q(\mathbf{y}, b)$ to represent $V_t^*(\mathbf{y})$, that is using the maximum value estimate of the randomly sampled state to be visited next in simulation as a proxy for the actual optimal value over the remaining planning period of the MDP model.

Pseudo-code of $Q$-learning is provided in Figure 3.4, for infinite-horizon MDP problems in general.



```
Initialise $Q(\mathbf{x}, a)$ to 0, for all $(\mathbf{x}, \mathrm{a})$. Initialise $k$ to 1, and set $k_{\max}$ to a
large integer. Pick an arbitrary state as the initial state.

while $k \leq k_{\max}$ do {
        1. choose action $a$ for state $\mathbf{x}$ (e.g. following Boltzmann
           selection rule)
        2. sample the state $\mathbf{y}$ to be visited next
        3. $Q(\mathbf{x}, a) \leftarrow (1 - \alpha)Q(\mathbf{x}, a) + \alpha(R(\mathbf{x}, \mathrm{a}) +$
           $\gamma \max_{b \in A(\mathbf{y})} Q(\mathbf{y}, b))$
        4. $k = k + 1$ and $\mathbf{x} = \mathbf{y}$
}

Compute and store $\pi(\mathbf{x}) = \arg\max_{a\epsilon A(\mathbf{x})} Q(\mathbf{x}, a)$ for any $\mathbf{x} \in S$

Return $\pi$ and $Q(\mathbf{x}, \pi(\mathbf{x}))$ for any $\mathbf{x} \in S$
```

Figure 3.4: Q-learning infinite planning-horizon [64]

### 3.3.3 Further technical discussions

#### Benefits of using Q-learning

Value iteration (VI) sweeps through the entire state-action space in each time-step (as explained in Section 3.2). Given a large state space, however, just one such sweep can render the computational time impractically long. Q-learning, however, selectively concentrates the computational efforts on some state-action pairs rather than visits all state-action pairs with the equal frequency (as explained in Section 3.3); intuitively speaking, in the application of Q-learning, the simulation would gradually more likely to sample state-action pairs that the MDP model would *actually* evolve into with relatively high probabilities following the optimal policy [64]. Thus Q-learning usually can converge in a relatively short computational time. In practice, the application of Q-learning can usually converge in a few hours while the application of VI would cost days.

Additionally, Q-learning only requires that the distributions of the individual governing random variables are obtained and stored, rather than the transition matrices of the system states as requested by VI, which reduces the modelling complexity and memory burden.

The discussion above in general also applies to other brute-force methods (such as policy iteration) and other reinforcement learning methods (such as SARSA).

**Optimality of Q-learning**

The following conditions are required to be fulfilled in order to ensure the estimate results of Q-learning converges *as* exact optimal results [149] : (1) all legit state-action(-time-step) pairs are sampled infinitely times; (2) $\sum_{k=1}^{\infty} \alpha_k = \infty$ and $\sum_{k=1}^{\infty} (\alpha_k)^2 < \infty$, where $\alpha_k$ denotes the step-size parameter value (in Equation 3.6 and Equation 3.9) after the $k^{th}$ sampling in the application of Q-learning. In other words, given the application of Q-learning contains infinitely large number of iterations (meaning infinitely long computational time) and given proper rules are used to control the step-size parameter and to sample state-action pairs (such as the rules discussed in Section 3.3.1), Q-learning is guaranteed to derive optimal results.

Of course no one can afford infinitely long computational time in practice, and additionally in our knowledge so far it is not proved that Q-learning (or any other reinforcement learning methods) derives *error-bounded* computational results in a *finite* computational time like VI; the mathematical proof from [149] however justifies the assumption that the converged results derived from proper application of Q-learning (meaning proper rules are implemented as aforementioned) in a *finite* computational time are relatively close to the exact optimal results. More specifically speaking, it is reasonable to assume such converged results contain relatively accurate estimates of both optimal values and optimal actions for the states that the MDP model would *actually* evolve into with relatively high probabilities following the optimal policy.

## 3.4 Value function approximation

This section dedicates to selecting an existing heuristic in response to the data-storage issue which arises from storing the computational results produced when applying a computation method to the large size MDP problem of the power plant case study (which would be discussed in Chapter 4).

As discussed above in this chapter, we shall choose Q-learning as the computation method for the power plant case study; the computational results (these are the $Q$ values in section 3.3.1 and section 3.3.2) should not be stored in brute-force lookup tables, otherwise the data-storage cost would be impractically large. In fact, for large size MDP problems in general, regardless of which computation method is applied, similar data-storage issues exist. Hence in this section we shall provide a brief summary of the main data-storage heuristics and justify why we choose *polynomial function method* for the power plant case study, in the hope that the heuristic selection criterion we use to facilitate our choice can serve as a ground rule for other researchers and practitioners to choose proper data-storage heuristics for their own studies.

In Section 3.4.1, we introduce different general heuristic approaches each of which bases on a unique methodology regarding how to tackle the general data-storage issue discussed above. Note hereafter in this section by “data” we mean the aforementioned computational results (which are the estimates of the expected values (defined in Section 3.2) of the state-action pairs in a given MDP problem). In Section 3.4.2, we further discuss specific heuristic methods developed following each such heuristic approach (from Section 3.4.1) and justify our choice of heuristic method for the power plant case study. Note here we aim at providing a holistic view without distracting readers from the main purpose of this thesis, and hence we keep the explanation of such existing heuristic approaches and heuristic methods relatively brief. Readers who are however interested in a more detailed discussion are referred to [129].

### 3.4.1 Different data-storage heuristic approaches

The main heuristic approaches aforementioned include the following:

- *State space clustering approach*: for each action of the MDP problem, such approach judges the relative closeness of the expected values of different states and group states deemed with close expected values into the same cluster. The approach only allows each cluster to store one representative value to approximate the expected values of all the states in the cluster. For a reasonable representation, the single representative value should be close to the mean of all the expected values of states in the cluster. Such approach uses brute-force lookup tables to store the representative values of all clusters for each action of the MDP problem. Here is a numerical example to illustrate such heuristic approach: suppose an MDP problem has two actions (indexed as action 1 and action 2) and six states (indexed as state 1,..., state 6); additionally, for action 1, state 1 and state 2 and state 3 are supposed to have close expected values while state 4 and state 5 and state 6 are supposed to have close expected values, whereas for action 2, state 1 and state 4 and state 6 are supposed to have close expected values while state 2 and state 3 and state 5 are supposed to have close expected values; as a result, for action 1, state 1 and state 2 and state 3 should be grouped into a cluster while state 4 and state 5 and state 6 should be grouped into another cluster, whereas for action 2, state 1 and state 4 and state 6 should be grouped as one cluster while state 2 and state 3 and state 5 should be grouped into another cluster.

- *Interpolation approach*: for each action of the MDP problem, such approach chooses exemplar states and allows their computational results to be explicitly stored. For each non-exemplar state, its expected value is interpolated from exemplars which are deemed to have relatively close expected values to the target non-exemplar state. The interpolation result is of course an estimate of the expected value of interest. Such approach uses brute-force lookup tables to store the computational results of the exemplar states for each action of the MDP problem. Here we reuse the numerical example above to illustrate the interpolation approach: for action 1, state 1 and state 3 can be selected as exemplars and their computational results can be used to interpolate the expected value of state 2, additionally state 4 and state 6 can also be selected as exemplars and their computational results can be used to interpolate the expected value of state 5; whereas for action 2, state 1 and state 6 can be selected as exemplars and their computational results can be used to interpolate the expected value of state 4, additionally state 2 and state 5 can also be selected as exemplars and their computational results can be used to interpolate the expected value of state 3.

- *Parametric function approximation approach*: for each action of the MDP problem, such approach requires to define a mathematical formula to approximate the numerical relationship between the states and their expected values. The resulted data-storage format of such approach is real valued functions and the value of each function is supposed to approximate the expected values of all the states for the corresponding action in the MDP problem.

In terms of the resulted data-storage format, the first two heuristic approaches above aim at reducing the total size of the brute-force lookup tables which would otherwise explicitly store the computational results for each state-action pair in the MDP problem, whereas the last heuristic approach aims at replacing brute-force lookup tables with real valued functions. In terms of methodology, both the *state space clustering approach* and the *interpolation approach* focus on judging the relative closeness of the expected values from different state-action pairs in an MDP problem, whereas the *parametric function approximation approach* focuses on the mathematical formula relationship between the state-action pairs and the expected values.

The specific heuristic methods in Section 3.4.2 inherits the methodology of the corresponding heuristic approaches above, with variance on technical choices.

### 3.4.2 Different data-storage heuristic methods

- Heuristic methods following the *state space clustering approach*: according to our best knowledge, the existing literature lacks such kind of heuristic methods that both have relatively good empirical performance (i.e. relatively small sacrifice of data accuracy) and can be potentially applied to a relatively wide range of case studies; an arguable exception is the so called *nearest neighbour method* which we shall specify below.

- Heuristic methods following the *interpolation approach*:

  - K-nearest neighbours method [42]: such heuristic method assumes the absolute difference of the expected values from two different states under the same action increases in proportion to the distance between the two states measured based on certain metrics: the most popular distance measuring choice is the so called Euclidean distance [48] and alternatively in some studies (for instance [65] ) researchers design problem-specific measuring techniques. For each non-exemplar state, $K$ exemplar states within the shortest distance (based on the measuring technique) to such non-exemplar state are selected. The expected value of the non-exemplar state is interpolated either (1) simply as the arithmetic mean of the computational results of all $K$ exemplar states or (2) interpolated as the weighted average of such $K$ computational results where the weight is based on the distance between the non-exemplar state and each corresponding exemplar state in a reciprocal numerical relationship or (3) interpolated from such $K$ computational results by regression [134]. An extreme case of the *$K$-nearest neighbours method* is when $K = 1$, and therefore only one exemplar state would be selected [56, 65]: no interpolation is actually applied and the method actually clusters states together to reduce the data-storage cost in a way which aligns with the methodology of the *state space clustering approach*; such extreme *$K$-nearest neighbours method* is referred as the *nearest neighbour method*.

  – Kernel-based method: such heuristic method is very similar to the *$K$-nearest neighbours method* above, and the main difference is here the expected value of every non-exemplar state is interpolated from the computational results of a fixed set of exemplar states by weighted averaging.

- Heuristic methods following the *parametric function approximation approach*:
  - Artificial neural networks (ANNs) [13]: such heuristic method uses a weighted sum of sigmoidal functions [33] or nested sigmoidal functions to approximate the numerical relationship of interest between the states and the expected values for each action in an MDP problem. According to studies including [33], with unlimited number of sigmoidal functions the numerical relationship of interest can be accurately captured.
  - Radial basis function networks (RBFNs) [21]: such heuristic method is very similar to the *ANNs method* above, and the only difference is here radial functions [22] are used rather than sigmoidal functions. According to [21], with unlimited number of radial functions the numerical relationship of interest can be accurately captured.
  - Polynomial function method: for each action of a given MDP problem, such

heuristic method uses a polynomial function to approximate the numerical relationship between the states and the expected values.

We use a general rule to select the data-storage heuristic for our case study: *the methodology and technical choices of the selected heuristic method should be backed up by the background knowledge of the case study.*

In our case study, the background knowledge implies that the numerical relationship between the system state and the expected value follows a concave pattern in general, which supports the choice of the *polynomial function method*: in more details, the system state is a high-dimensional variable which is a vector comprised of multiple basic variables (i.e. the condition/performance of each individual mill in the power plant); according to the background knowledge of our case study, the condition/performance of a mill impacts the expected value via determining the lifetime/output-rate of the mill, and further numerical test shows the numerical relationship between the condition/performance and the average lifetime/output-rate follows concave patterns in the case study. We shall specify such discussion and explain how to apply the *polynomial function method* to our case study later in Chapter 4, and additionally in Chapter 4 we shall also specify how the *polynomial function method* interacts with Q-learning.

As comparative examples (which follow the same heuristic selection rule as ours), in both an automatic guided vehicle route scheduling case study [134] and a video gaming strategy developing case study [56], the system state consists of the physical location of objects; the background knowledge in such studies implies the physical distance between two system states determines the relative closeness of the expected values of such system states in the MDP models, and hence it is reasonable to adopt the *(K-)nearest neighbour(s) method* and use Euclidean distance in such studies. In a yield management case study [65] from the airline industry, the system state consists of multiple basic variables which describe the historical flight classes booking situation (including the number of seats booked in each flight class and the booking time of each seat) and the current booking request; the background knowledge in the case study suggests the relative closeness of the expected values of such system states which have identical current booking request can be approximately judged by the amount of already earned revenue from booked seats; hence [65] adopts the *nearest neighbour method* and additionally develop a revenue index function (such function prescribes the normalised earned revenue based on booked seats and the cost/benefit structure in the case study) to judge the relative closeness of system states.

We would also like to highlight that the heuristic methods discussed above with universal-approximation power (i.e. the *ANNs method* and the *RBFNs* method) are relatively less favoured by our heuristic selection rule aforementioned, as such heuristics only provide a relatively limited gateway for researchers/practitioners to link the *parametric function approximation* modelling choices to the specific background knowledge of their case studies: regarding such limited gateway, interested readers are referred to publications including [13, 64] for discussions about how to first construct some simple functions of the state variable and then use such simple functions as the independent variables of the sigmoidal functions in the *ANNs method*; such simple functions are referred to as features in such publications and in theory they do provide a gateway which links to the problem-specific background knowledge; the fundamental sigmoidal/radial functions based real-value function structure in such methods are nonetheless not adjustable based on the problem-specific background knowledge at all. We admit that using such heuristics with universal-approximation power is prevalent so far in the research area and relatively good (meaning the data accuracy sacrifice is relatively small) empirical results are reported in multiple studies including [40, 72, 143]; however researchers/practitioners can expect two (usually impractical) requirements in general about applying such heuristics: (1) a relatively large amount of set-up efforts/time on constructing the so called features aforementioned, and such issue is reported in multiple publications (e.g. [13, 40, 64, 72, 118, 151]) including the ones with relatively good empirical results ; (2) a relatively large amount of free-parameters tuning efforts/time induced by adopting multiple sigmoidal/radial functions: usually complex decision making problems in practice lead to large size MDP problems in which the numerical relationship between the states and the expected values is highly non-linear, and hence multiple sigmoidal/radial functions are required for a relatively close approximation of such numerical relationship of interest. In contrast, our heuristic selection rule above encourages researchers/practitioners to identify a problem-specific data-storage method which hopefully only induces a relatively small amount of data accuracy sacrifice without imposing an impractically high demand on the set-up time and/or free-parameters tuning efforts. Additionally, we would like to highlight that our heuristic selection rule does not aim at discouraging researchers from continuing with *theoretical* research related to such universal-approximation power methods or stopping researchers/practitioners who have access to high-performance computing devices from applying such methods to case studies which are underpinned by the purpose of exploring universal artificial intelligence (such as [75] and [130] ).

In Chapter 4, first we shall apply the performance-centred approach (from Chapter 4) to the power plant case and model the maintenance planning decision making problem for the case study; then we shall design a set of heuristics to solve the corresponding mathematical problem. The *polynomial function method* provides a relatively general framework and an important part of our own design of heuristics in Chapter 4 is specifying the application of such general method in our case study. Both the mathematical model and the heuristics in Chapter 4 apply to a relatively general context and therefore can be used on other cases as well (we shall specify the discussion in Chapter 4).

# Chapter 4

# Case study: modelling and heuristics

In this chapter, we apply the performance-centred maintenance approach from Chapter 2 to the coal-fired power plant that motivates our study. The power plant is one example of many production companies from various industries where the production operations management and maintenance operations management should be aligned in order to achieve better total costs/benefits (see Section 2.1 for specifications). In addition, the power plant presents a general challenge in terms of applying the performance-centred maintenance approach: the production machines in the power plant forms a multi-level hierarchical physical structure (which we shall specify in Section 4.1.3) which is beyond the scope of the system-asset physical structure (see Section 2.1) modelled following the performance-centred maintenance approach. Such hierarchical structure is shared by a vast number of industries (which we shall discuss in Section 4.1.3) and aligning the production and maintenance operations management in such industries necessitates scaling up the existing maintenance planning mathematical model (developed in Chapter 2) under the performance-centred maintenance approach. Therefore, building such widely applicable scaled-up mathematical model is the first aim of this chapter.

The resulted mathematical model is very complex, and thus heuristics are required to solve the mathematical problem, which gives rise to the second aim of the cases study:

developing a set of heuristics which can tackle the mathematical problem that arises from aligning the production and maintenance operations management in industrial problems which share the same hierarchical structure as the power plant.

Below in Section 4.1 we introduce the power plant case study background information and then formally describe the hierarchical structure and the case study; in Section 4.2 we develop the scaled-up mathematical model in the context of the power plant, and discuss how such model can be applied to other cases of interest; in Section 4.3 we develop a set of heuristics to solve the mathematical problem in the context of the power plant, and discuss how such heuristics can be applied to other cases of interest; in Section 4.4 we summarise the research contributions of this chapter.

## 4.1 Case study: problem structure and assumptions

Below in Section 4.1.1, we shall describe the problem setting of the power plant case study and highlight the relevance of the performance-centred maintenance approach; in Section 4.1.2, we describe the operators' decision regimes in the context of the case study which we shall numerically investigate and compare in Chapter 5; in Section 4.1.3, we specify the hierarchical structure shared by the power plant case study and various other production companies; finally, in Section 4.1.4 we formally describe the power plant case study.

### 4.1.1 Background of the coal-fired power plant

The case study is based on the real-life maintenance problem in a coal-fired power plant, which consists of four separate independent units, each of which contains eight mills working in parallel (such multi-level structure is illustrated in Figure 4.1). The specific physical structure of a mill is illustrated in Figure 4.2. Without further discussing the multiple elements in each mill, the production operations from a high level can be explained as follows: for each unit, the operators regulate the speed by which the coal is fed to the mills in the unit, and the mills grind the coal to coal dust which is transferred into the boiler of the unit to be burned in order to power the steam turbines and generate electricity. The focus is on the operators' utilisation behaviours and maintenance policy for the mills rather than the boiler system or the turbines.

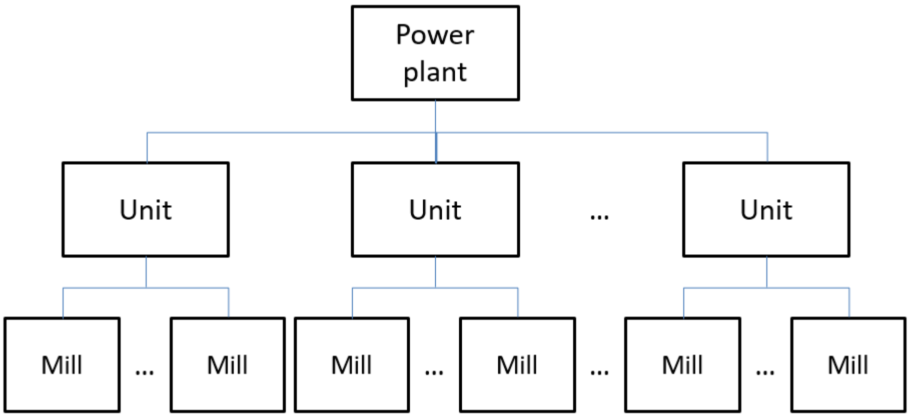


Figure 4.1: Hierarchical structure in the power plant

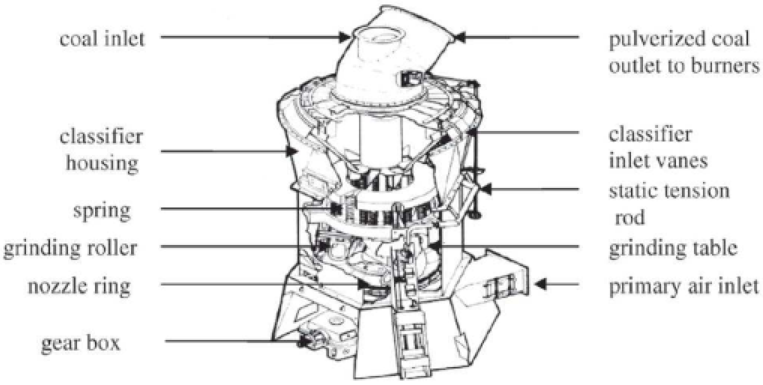


Figure 4.2: Mill structure [4]

The performance of a mill affects the maximum amount of coal a mill can process per unit of time (that is the *throughput* of a mill) and the particulates size distribution of the coal dust transferred to the boiler (that is the *grinding quality* of a mill). The amount of electricity generated is determined by both the quantity and quality of coal grinding. The condition of a mill determines the likelihood of a critical failure and the state of wear-out of the mill. According to the maintenance engineers, the performance of a mill will generally deteriorate as the condition of the mill deteriorates; however, there are occasions when this pattern is not followed. A service maintenance action can improve the performance of a mill with a relative short maintenance time and low fixed cost, but it has no effects on the degradation condition of a mill and therefore there is no effect if the mill has failed in terms of degradation condition. Services on the mills include tasks such as rollers oil changes, rollers tension adjusting and others. An overhaul maintenance action can restore both the performance and condition of a mill to the full levels, but with a relatively long maintenance time and high fixed cost. The examples of condition failures for a mill include cracking of the grind plate, wear of the rollers below a useful operational standard and wear of structural elements beyond certain thresholds. A service consumes approximately one week and an overhaul takes approximately five weeks. It is rare for maintainers to pull offline more than one mill for maintenance at any given time in a unit, due to the concern of overly reducing the production capacity of the unit, which is based on the observed average electricity supply level contracted in history (note this is a rule of thumb that maintainers follow in practice rather than a definite requirement, and it only applies to mills which are still working rather than the mills which are not contributing to production due to condition/performance failure). Additionally, due to financial and personnel restrictions, maximum two maintenance crews are available across the plant at a given time: one performs overhaul and one performs service. Each crew consists of a number of staff. Usually the overhaul of a mill requires a full overhaul crew and the service of a mill requires a full service crew. The maintainers' business function is planning maintenance for the mills and applying scheduled and unscheduled maintenance actions, and their focus is ensuring that the whole power plant is available as required for production in a long term.

In contrast, the operators are relatively short-term oriented, due to the existence of the contracted period: the electricity is sold by contracts for a relative short period in advance, in terms of both quantity and price. According to [135], on the British electricity wholesale market, “contracts for electricity can be struck over timescales ranging from several years ahead to on-the-day trading markets” and the production is matched with the contracted supply on a second-by-second basis as electricity cannot be stored in large amounts. Large scale storage of electricity is either impossible or very expensive; hence if the production cannot satisfy the contracted supply, any missing amount must be procured from the spot market which requires a potentially substantial financial premium compared to self-production. To avoid/mitigate procurement on the potentially expensive spot market when the electricity generation cannot satisfy the contracted supply under normal production circumstances, the operators can choose to speed up production in the working mills to a level that is above the usual level but such behaviour may accelerate deterioration of the mills (we shall discuss the operators’ decision regime in more details below in Section 4.1.2). The operators in the power plant adopt such machine utilisation behaviours because their business function is planning and executing production, and they focus on meeting the contracted supply while having limited interests in the effects that their decision regime brings to the relative long-term state of the mills.

Despite the difference in the focus of operations management between the maintainers and the operators, the business functions of the two managerial parties are intertwined in a complex way: the scheduled maintenance actions can improve the machine state and therefore lead to a better production performance in terms of quantity and/or quality, but during the maintenance some loss of production may happen in addition to fixed maintenance costs, and the trade-offs differ especially significantly between services and overhauls which are operational performance oriented and degradation condition oriented respectively; due to potentially higher cost on the spot market, the operators are motivated to speed up production when electricity generation at normal rate cannot satisfy the contracted demand, but such machine utilisation behaviours render machine deterioration faster, which may impose higher necessity for maintenance in the future.

As a result, maintenance operations management and production operations management should not be considered independently in the power plant. Instead, the business functions of the two managerial parties should be aligned in order to achieve optimal total benefits/costs at the plant level, not least because the interests between the operators and maintainers may conflict as priority needs to be set between short-term production and long-term health of the mills.

A similar requirement for such alignment applies to many industrial cases in which the operations management is split between the operators and maintainers. The maintenance approaches in the existing literature are however largely reliability-centred, which is not able to model the complex trade-offs that arise from aligning the business functions of the two managerial parties. Therefore in this thesis we build a new performance-centred maintenance approach (based on [7, 8]) in Chapter 2, in order to break down the decision making barrier between the maintainers and the operators for such production systems where the two managerial parties follow different timescales. Such maintenance approach is widely applicable given its generic business setting (discussed in Section 2.1) and the fact that the benefits and costs involved in the decision making can usually be evaluated in monetary value for many production companies.

### 4.1.2 Operators' decision regime in power plant

Based on the investigation by [7], the existing operators' decision regime in the power plant can be approximately summarised as follows: every unit in the power plant is committed to a separate contract, and if a unit cannot meet the production target under normal operation circumstances, then the operators would speed up the work-rate of all the working mills in the unit to a level higher than normal. Such operator decision regime results in stochastic dependence between the residual lifetimes of the mills in the same unit. Note that any production shortfalls of a unit must be resolved by emergent procurement from the spot market, rather than by exploiting the potential extra production capacity of other units in the plant: such setting results from the specific requirements of the British electricity wholesale market and we shall not further discuss the details.

Note in practice the operators actually have more flexible choices, we choose the approximation above in order to avoid over-complicating the mathematical model that we shall build later in this chapter such that we can focus on the key trade-offs between the maintenance operations management and production operations management in the case study. This thesis serves as an early-stage study of its kind, and in the future research we plan to incorporate more advanced decision regimes to better capture the real machine utilisation behaviours of the operators in the power plant: for example, rather than adjust the work-rate at the unit-level, the operators may further adapt the work-rate of each individual mill and only increase the work-rate of the mills in relative good states. In this thesis, we refrain from any further discussion about more advanced decision regimes which involve work-rate adjustment at the mill level or the plant level.

Hereafter in this thesis, we only consider work-rate adjustment at the unit level. In such operators' decision regime, the operators have the choice to speed up work-rate but as a consequence the degradation may also be accelerated. It is worth investigating whether such trade-off is beneficial, and therefore an alternative decision regime shall be examined which prohibits the operators from speeding up the production. This alternative regime induces a relatively low level of intervention to the maintenance planning, compared to the original regime. Hereafter in this thesis, we refer to the alternative regime as the *low intervention regime* and to the original one as the *high intervention regime.*

### 4.1.3 Hierarchical structure in general

The power plant maintenance planning problem in essence requires decision making about maintenance resources distribution at three sequential levels: at the highest plant level, the maintainers need to decide which unit in the plant should be maintained; furthermore, at the medium unit level, the maintainers should decide which mill in a unit should be maintained; finally, at the bottom mill level, the maintainers have to decide whether to apply service or overhaul to a mill. Generally speaking, each mill in the power plant is a production asset, each unit in the power plant is a production system which consists of multiple production assets, and the power plant itself is a company that is comprised of several production systems. Hereafter by *hierarchical structure*, we mean such general company-system-asset structure, as illustrated in Figure 4.3. The multi-level physical structure (illustrated in Figure 4.1) of the power plant is one example of such hierarchical structure.

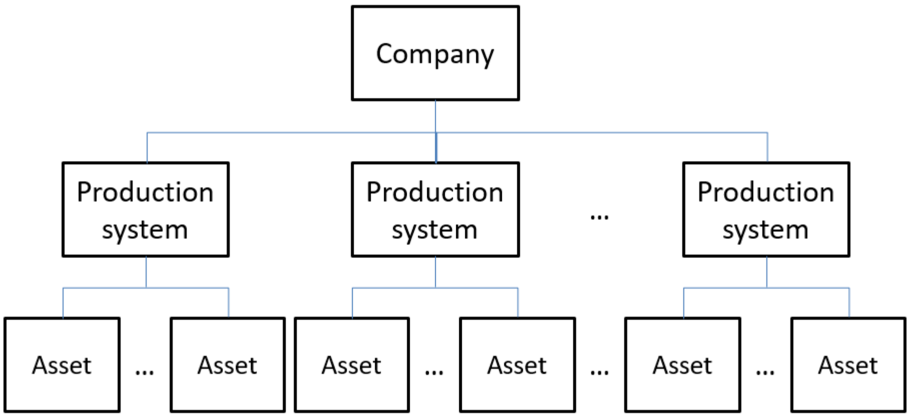


Figure 4.3: General hierarchical structure

It is actually not exclusive to the coal-fired power plant that the maintenance resources distribution follows such hierarchical structure. Various examples can be found in industries that have adopted distributed generation or distributed manufacturing, including the solar power industry, wind power industry, automotive manufacturing sector, electronics manufacturing sector and healthcare manufacturing. For instance, think of a car manufacturing company which has five factories and each factory is comprised of ten production lines and additionally the maintenance resources are shared at the company level: in such case each production line can be seen as a production asset and each factory can be seen as a production system and therefore the maintenance resources distribution again follows the company-system-asset hierarchical structure.

Such a hierarchical structure poses a crucial challenge to applying the performance-centred maintenance approach: in the performance-centred maintenance approach (from Chapter 2), the production machines are assumed to follow a system-asset structure, which does not further include the company level as considered in the hierarchical structure here; in other words, the multi-level hierarchical structure here is beyond the scope of the mathematical model built in Chapter 2. As a result, such model should be scaled up for more complex maintenance planning problems which follow such hierarchical structure.

Below we apply the performance-centred maintenance approach (from Chapter 2) to the the coal-fired power plant case study and build a case-specific mathematical model. Additionally, we shall also build a set of heuristics to solve the corresponding mathematical problem. The mathematical model and the set of heuristics can be relatively easily adopted to many other industries which require the alignment of the production and maintenance operations management under the same hierarchical structure discussed above.

### 4.1.4 Case study

In this section we formally describe the power plant case study.

The whole power plant consists of $K$ units which are indexed as $1, 2, ..., K$ respectively, and each unit contains $N$ identical mills working in parallel which are indexed as $1, 2, ..., N$ respectively in a given unit. The mills deteriorate stochastically with production. The performance of an mill is improved after a service, given the asset is not failed in terms of condition. The condition and performance of a mill are fully restored after an overhaul. Overhaul consumes a longer time and induces a higher cost than service, as an overhaul action is much more complex than a service action. The operators adjust the work-rate of each unit based on (1) the contracted supply and (2) the unit state, by following either the *high intervention regime* or the *low intervention regime* specified in Section 4.1.2; the maintainers are grouped into two different crews: one crew is dedicated to service and the other crew is dedicated to overhaul, and the maintainers need to plan maintenance actions up to a certain horizon. Each crew can only maintain one mill each time. Large scale of electricity storage is either impossible or very expensive (given the current technology) such that inventory would never be an optimal choice and therefore it is not considered.

Additionally, here we introduce some modelling assumptions in order to focus on the key trade-offs in the case study:

- The service of a mill consumes one time-step and the cost of service is fixed; the performance of a mill is fully restored after a service, as long as the mill is not failed in terms of condition.
- The overhaul of a mill consumes $OH$ time-steps and the cost of overhaul is fixed.
- The condition of a mill deteriorates gracefully with time such that the condition only degrades by up to one level in a single time-step. Following such an assumption, one can derive the transition probabilities between any two condition levels based on the estimated average operating time a mill spends at each condition level at each discrete level of work-rate.

**Simplified numerical examples: PCM approach, myopic approach and other existing maintenance approaches**

Here we take a detour to discuss a highly simplified toy-size problem based on the business context of the real-life case study above, aiming at numerically demonstrating the value of adopting the more sophisticated PCM approach in comparison to (1) the myopic approach (discussed in Section 2.2.4.3) which only captures the contracted-period and other existing maintenance approaches (discussed in Section 1.2 and Section 2.3) which (2.a) assume the condition and performance are perfectly correlated and the ones (2.b) *further* assume that production has deterministic wearing-off effects on machines. Below in Section 4.2 we shall return to the full-size case study.

In this toy-size problem, we assume the following: (1) the power plant contains only 1 unit which has 3 identical mills; (2) the contracted period contains 20 weeks; (3) the entire maintenance planning period contains 50 weeks; (4) the demand in each week within the non-contracted period (which is 30 weeks) follows the same continuous uniform distribution unif$(a, b)$ where $a$ equates to 70% of the maximum weekly production rate of a perfect unit working at normal work-rate and $b$ equates to 100% of such maximum weekly production rate, while the demand in each week within the contracted period is equal to 90% of such maximum weekly production rate; (5) the operators follow the *low intervention regime* (note in this simplified toy-size problem we shall not further investigate whether *high intervention regime* is more profitable); (6) although the condition deterioration underpins the performance deterioration, the performance deterioration is not perfectly linked to condition deterioration; (7) service and overhaul cannot be applied to the unit at the same time (note this assumption is introduced to simplify the case studies here and it is based on the rule of thumb aforementioned in Section 4.1.1).

Below we shall first present the mathematical model and the derived decisions for PCM approach as a benchmark; then we shall use the benchmark to highlight the limitations of the mathematical models and the decisions derived based on the other aforementioned approaches[1]. Note here we use the *value iteration* computation method (see Section 3.2) and store the computational results in brute-force lookup tables when deriving the decisions from different mathematical models, in order to conduct the comparison between different modelling approaches based on accurate computations.

PCM approach: mathematical model and derived solutions

As a general modelling choice, the condition of a mill is categorised into four levels (new, good, poor and failed) and the performance of a mill is categorised into three levels (full performance: that is satisfactory performance; reduced performance: that is unsatisfactory performance; offline: that is grossly unsatisfactory performance); we further

[1] The compressed MATLAB code for all the toy-size case studies can be found at http://doi.org/10.5281/zenodo.2602868

index all potential mill state and describe the corresponding production rate as shown in Table 4.1 in order to facilitate later discussion. In addition, a time-step is measured as one week.

| **Mill state index** | **Meaning (condition, performance)** | **Weekly production rate (in terms of standard units of demand) at normal work-rate** |
|---|---|---|
| 1 | (failed, offline) | 0 |
| 2 | (poor, offline) | 0 |
| 3 | (poor, reduced performance) | 60 |
| 4 | (poor, full performance) | 80 |
| 5 | (good, offline) | 0 |
| 6 | (good, reduced performance) | 60 |
| 7 | (good, full performance) | 80 |
| 8 | (new, offline) | 0 |
| 9 | (new, reduced performance) | 60 |
| 10 | (new, full performance) | 80 |

Table 4.1: Different mill states

The mathematical model for PCM approach follows the format defined in Equation (2.3) (see Section 2.2.4.3). Here we specify the parameter values of the model: the state space contains 220 unique states at the unit level (see Table A.1 in Appendix A.1 for specification); the probability of state transition at the individual mill level is given in Table 4.2, and the mill state transition diagram is given in Figure 4.4; the maintenance choices at the individual mill level are listed and indexed in Table 4.3; the time length and the cost of *service* and *overhaul* are given in Table 4.2; the sales price and the penalty cost per unit of demand are also given in Table 4.2. Note here the presented parameter values are desensitised (meaning the business sensitive parameter values derived from the original real-life case are modified without losing the dynamic nature of the problem, and then presented here to readers).

The derived decisions (on maintenance choice) and expected value for each potential *initial* unit state is given in Table A.2 in Appendix A.1. For example, for initial unit

| Parameter | Meaning | Parameter value |
|---|---|---|
| $P_M$ | Mill state transition matrix under normal work-rate and no maintenance | $\begin{bmatrix} P_{1,1} & P_{1,2} & ... & P_{1,10} \\ P_{2,1} & P_{2,2} & ... & P_{1,10} \\ \vdots & \vdots & \vdots & \vdots \\ P_{10,1} & P_{10,2} & ... & P_{10,10} \end{bmatrix} = \begin{bmatrix} 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 \\ \vdots & \vdots & \vdots & 0 \\ 0 & 0 & ... & 0.93765 \end{bmatrix}$ |
| $C_{OM}$ | cost of overhaul (£) | $C_{OM} = 207,480$ |
| $C_{SM}$ | cost of service (£) | $C_{SM} = 1,920$ |
| $T_C$ | time length of contracted period (weeks) | $T_C = 20$ |
| $T$ | time length of maintenance planning period (weeks) | $T = 50$ |
| $OH$ | time length of overhaul (week) | $OH = 1$ |
| $SH$ | time length of service (week) | $SH = 1$ |
| $R_S$ | sales price (per unit of electricity) (£) | $R_S = 3,780$ |
| $R_P$ | penalty price (per unit of electricity) (£) | $R_P = 1,134$ |

Table 4.2: PCM approach: parameter values

state 1, maintainers are recommended to apply no maintenance; the expected value (£) is 27561445.32.

Myopic approach: mathematical model and derived solutions

The mathematical model derived based on the myopic approach is similar to the mathematical model derived above based on the PCM approach (and we choose not to repeat the parameter values here), with the only difference here the maintenance planning period $T = 20$ ($T = 50$ above in PCM approach).

For the myopic approach, the derived decisions (on maintenance choice) and the *loss* of expected value (compared with the optimal decision derived from PCM approach) for each potential *initial* unit state are listed in Table A.3 in Appendix A.2. For example, for initial unit state indexed as 211 (see Table A.1 in Appendix A.1 for the meaning of the unit state indexing), the myopic approach recommends maintainers to apply no maintenance to any mill but the optimal choice actually would be applying overhaul to

| **Maintenance choice index** | **Meaning** (see Table 4.1 for reference to mill state index) |
|---|---|
| 1 | Applying no maintenance to any mill in the unit |
| 2 | Applying service to a mill in mill state 2 |
| 3 | Applying service to a mill in mill state 3 |
| 4 | Applying service to a mill in mill state 5 |
| 5 | Applying service to a mill in mill state 6 |
| 6 | Applying service to a mill in mill state 8 |
| 7 | Applying service to a mill in mill state 9 |
| 8 | Applying overhaul to a mill in mill state 1 |
| 9 | Applying overhaul to a mill in mill state 2 |
| 10 | Applying overhaul to a mill in mill state 3 |
| 11 | Applying overhaul to a mill in mill state 4 |
| 12 | Applying overhaul to a mill in mill state 5 |
| 13 | Applying overhaul to a mill in mill state 6 |
| 14 | Applying overhaul to a mill in mill state 7 |
| 15 | Applying overhaul to a mill in mill state 8 |
| 16 | Applying overhaul to a mill in mill state 9 |

Table 4.3: PCM approach: action space

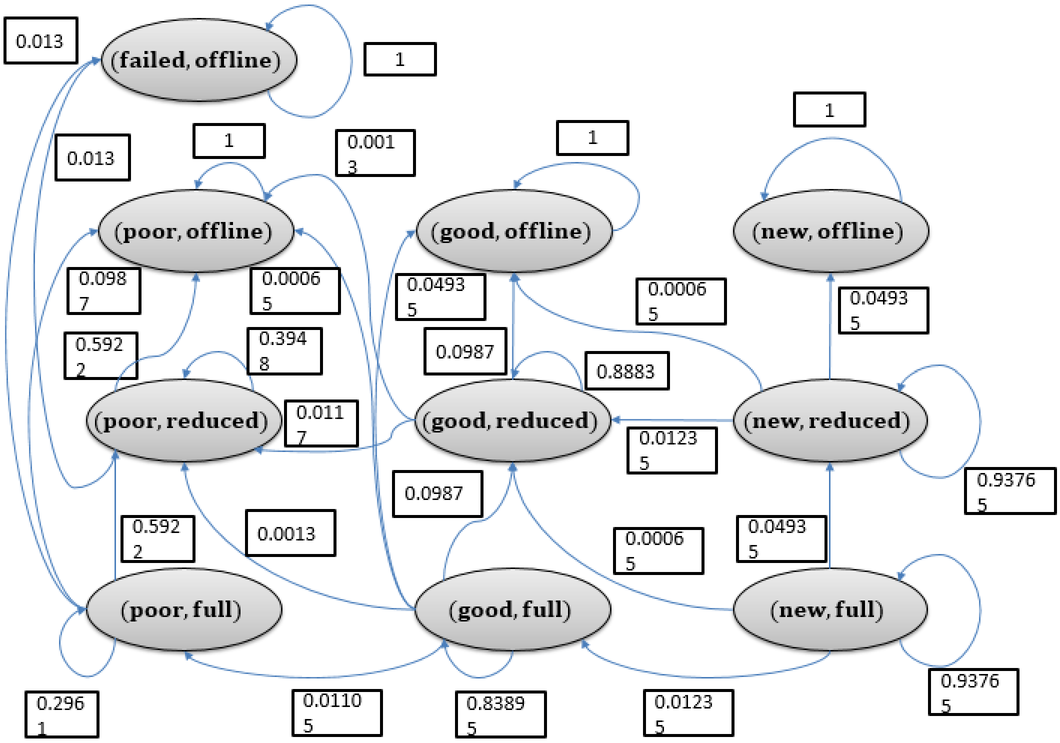


Figure 4.4: Mill state (condition, performance) transition diagram

a mill that is in the mill state indexed as 1 (see Table 4.1 above for the meaning of the mill state indexing); the expected loss of value is 38.07% (based on the expected values of the two maintenance choices and the expected values are both computed from the model derived based on the PCM approach).

As shown in Table A.3, compared with the PCM approach, the myopic approach shows a strong tendency to avoid costly but actually necessary maintenance choices (for some initial unit states, the optimal choice of service is replaced by no maintenance, and the optimal choice of overhaul is replaced by either service or no maintenance), and the expected loss of value ranges from 0% to 39.96% for the potential *initial* unit states. Deriving low quality sub-optimal decisions (as illustrated here in the toy-size problem) can be a common issue for the myopic approach, as (discussed in Section 2.2.4.3) such approach usually cannot effectively capture the full benefits of expensive (in terms of monetary value and/ or time cost) maintenance choices.

Perfect correlation approach: mathematical model and derived solutions

The same as in the PCM approach, here the condition of a mill is categorised into four levels and the performance of a mill is categorised into three levels (readers can refer to the meaning of each such categorised level above in the PCM approach). However, following the basic assumption "condition and performance are perfectly correlated", the perfect correlation approach is limited to coarsely modelling the toy-size problem as if the performance deterioration is perfectly linked to condition deterioration. In other words, in the mathematical model here, each condition level can only be associated with one performance level, rather than multiple performance levels. We further index all potential mill state and describe the corresponding production rate as shown in Table 4.4.

Here we still choose to build the mathematical model by following the format defined in Equation (2.3) (see Section 2.2.4.3). Note there exist other (perhaps more succinct) modelling formats for the perfect correlation approach, and our choice here of using the same format as defined in Equation (2.3) aims at helping readers focus on the loss of modelling accuracy if one chooses to follow the perfect correlation approach (in comparison to the PCM approach) for the toy-size problem.

Here we specify the parameter values of the model: the state space contains 20 unique states at the unit level (see Table A.4 in Appendix A.3 for specification); the probability of state transition at the individual mill level is given in Table 4.5; the maintenance choices at the individual mill level are listed and indexed in Table 4.6; the values of other parameters (the time length and the cost of *overhaul*, the sales price and the penalty cost per unit of demand) remain the same as PCM approach above in Table 4.2 and we choose not to repeat them here in Table 4.5. Note as illustrated by the difference between 4.3 above from PCM approach and Table 4.6 here, the perfect correlation approach cannot effectively model maintenance choices that are mostly effective at improving performance but not effective on condition (these maintenance choices are *services* in this toy-size problem). In other words, the coarse mathematical model derived based on the perfect correlation approach risks overlooking an important type of maintenance choices from real-life, which is a common issue to the perfect correlation approach.

| **Mill state index** | **Meaning (condition, performance)** | **Weekly production rate (in terms of standard units of demand) at normal work-rate** |
|---|---|---|
| 1 | (failed, offline) | 0 |
| 2 | (poor, reduced performance) | 60 |
| 3 | (good, full performance) | 80 |
| 4 | (new, full performance) | 80 |

Table 4.4: PC approach: different mill states

| **Parameter** | **Meaning** | **Parameter value** |
|---|---|---|
| $P_M$ | Mill state transition matrix under normal work-rate and no maintenance | $\begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{1,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.013 & 0.987 & 0 & 0 \\ 0 & 0.013 & 0.987 & 0 \\ 0 & 0 & 0.013 & 0.987 \end{bmatrix}$ |

Table 4.5: PC approach: parameter values

| **Maintenance choice index** | **Meaning** (see Table 4.4 for reference to mill state index) |
|---|---|
| 1 | Applying no maintenance to any mill in the unit |
| 2 | Applying overhaul to a mill in mill state 1 |
| 3 | Applying overhaul to a mill in mill state 2 |
| 4 | Applying overhaul to a mill in mill state 3 |

Table 4.6: PCM approach: action space

For the perfect correlation approach, the derived decisions (on maintenance choice) and the *loss* of expected value (compared with the optimal decision derived from PCM approach; note the comparison is conducted based on mapping the coarse state space and coarse action space of the perfect correlation approach with the ones above from the PCM approach) for each potential *initial* unit state are listed in Table A.5 and Table A.6 respectively in Appendix A.3. For example, for initial unit state indexed as 9 (in the state space of PCM approach) the perfect correlation approach recommends applying no maintenance to any mill but the optimal choice would actually be applying service to a mill in mill state 8 (in the mill space of PCM approach; see Table 4.1 for the meaning of the mill state indexing); the expected loss of value is 1.98%.

Based on Table A.6 in Appendix A.3, the maintenance policy derived from the perfect correlation approach seems to relatively well approximate the optimal policy from the PCM approach: the expected loss of value ranges from 0% to 4.41% for the potential *initial* unit states, despite the perfect correlation approach completely overlooks an important type of maintenance choices. However, such relatively good approximation effects of the perfect correlation approach is confined to the specific parameters setting in this toy-size problem; in other words, the perfect correlation approach may yield much lower quality decisions in other problems. Future studies can conduct more in-depth sensitive analysis on the input parameters in Table 4.2.

Deterministic wearing-off approach: mathematical model and derived solutions

Compared with the perfect correlation approach above, here the deterministic wearing-off approach further assumes that production has deterministic wearing-off effects on machines. In more details, given that the mill condition is categorised into several discrete levels (rather than modelled as a continuous variable) in this toy-size problem, the mill condition is assumed to stay at the same level for a fixed time period with probability 1 and then the mill condition would deteriorate by one level immediately after the fixed time period with probability 1 (note this means the mathematical model derived here would be even more coarse compared to the one derived based on the perfect correlation approach). Furthermore, given the "graceful condition deterioration" assumption above in Section 4.1.4, the aforementioned fixed time period is the same for each mill condition level.

The mathematical model derived based on the deterministic wearing-off approach is similar to the mathematical model derived above based on the perfect correlation approach (and we choose not to repeat the parameter values here), with the only differences here (1) an extra parameter $T_X$ is introduced to denote the aforementioned fixed time period, and here $T_X = 78$ (that is 78 weeks; note in other problems the value of $T_X$ may vary between different condition levels) and (2) the mill state transition matrix is updated as shown in Table 4.7.

| Parameter | Meaning | Parameter value |
|---|---|---|
| $P_{M1}$ | Mill state transition matrix under normal work-rate and no maintenance (before reaching $T_X$) | $\begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{1,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ |
| $P_{M2}$ | Mill state transition matrix under normal work-rate and no maintenance (when reaching $T_X$) | $\begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{1,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ |

Table 4.7: Deterministic wearing-off approach: parameter values

Regarding the deterministic wearing-off approach, the derived decisions (on maintenance choice) and the *loss* of expected value (compared with the optimal decision derived from PCM approach; note the comparison is conducted based on mapping the coarse state space and coarse action space of the deterministic wearing-off approach with the ones above from the PCM approach) for each potential *initial* unit state are listed in Table A.7 and Table A.8 respectively in Appendix A.4. For example, for initial unit state indexed as 38 (in the state space of PCM approach) the deterministic wearing-off approach recommends applying no maintenance to any mill but the optimal choice actually would be applying overhaul to a mill in mill state 5 (in the mill space of PCM approach; see Table 4.1 for the meaning of the mill state indexing); the expected loss of value is 2.19%.

As shown in Table A.8 (in Appendix A.4), the expected loss of value for the deterministic wearing-off approach ranges from 0% to 9.94% for the potential *initial* unit states, which is worse compared with the results from the perfect correlation approach above. Furthermore, by comparing Table A.8 (in Appendix A.4) with Table A.6 (in Appendix A.3), the deterministic wearing-off approach shows a stronger tendency to avoid beneficial preventive maintenance choices compared with the perfect correlation approach, which can be a common issue for deterministic wearing-off approach: because the approach additionally assumes the machine system would not deteriorate for a fixed time period, the resulted maintenance policy would be more likely to avoid beneficial preventive maintenance choices during the fixed time period.

Conclusion of toy-size problem case studies

In this section we compare the numerical performance between different approaches in a simplified toy-size problem. As demonstrated in the numerical tests, the more sophisticated PCM approach helps derive a more accurate mathematical model and a better quality (in terms of total costs/benefits) maintenance policy compared with the the other approaches.

Note in this simplified toy-size problem we only compare the numerical performance between different approaches on maintenance decision making, and their performance on choosing the operators' decision regimes is not tested yet. It is reasonable to assume the PCM approach would also yield better results than the other approaches on choosing the operators' decision regimes, and more advanced case studies can be conducted in the future research to test the assumption.

## 4.2 Three-stage hybrid MDP model

In this section we apply the performance-centred maintenance approach from Chapter 2 to the full-size case study described above in Section 4.1. The essence of such application is scaling up the existing mathematical model of the maintenance approach, as previously discussed in Section 4.1.3.

The optimal decision making for the case study (see Section 4.1.4) is potentially very complex, and therefore here we additionally combine the performance-centred maintenance approach with a heuristic decomposition method in the modelling work for the case study, in order to mitigate the complexity of the resulted mathematical model. Below in Section 4.2.1 we explain the main idea of the decomposition method and the framework about how to integrate such decomposition method into the performance-centred maintenance approach, and additionally we discuss how to adapt the decomposition method for other cases of interest; then in Section 4.2.2 we present the mathematical model.

### 4.2.1 Decomposition method and modelling framework

In this study we focus on maintenance planning problems with an infinite planning-horizon, and the resulted mathematical model can be easily modified for finite planning-horizon problems. Under our reconstructed performance-centred maintenance approach (see Chapter 2), the infinite maintenance planning period is split into two parts: contracted period and non-contracted period (as discussed in Section 2.2.1), the time-lengths of which are finite and infinite respectively. The decomposition method applies to the non-contracted period.

Below we first focus on decomposing the maintenance decision making in the non-contracted period, and then we discuss how to integrate the decomposition method into the performance-centred maintenance approach to model the complete maintenance planning problem.

#### 4.2.1.1 Two-timescale decomposition method

For the non-contracted period, we reduce the complexity of optimal decision making for the maintainers by further splitting the decision making into two parts: short-term part and long-term part. For the short-term part of the non-contracted period, we shall still accurately model the decision making problem; for the rest (long-term part) of the non-contracted period, we only consider and model high-level maintenance decision making by assuming the maintenance resources are distributed evenly in the power plant at both the unit level and the mill level in the long-term.

**Deterministic rotating schedules**

More specifically speaking, the overhaul crew is assumed to operate on a deterministic rotating schedule such that (1) from the perspective of the unit level, the overhaul crew visits all the units in the power plant one by one in turn repeatedly and (2) from the perspective of the mill level, the overhaul crew visits all the mills in the power plant one by one in turn repeatedly. Here is a numerical example of such deterministic rotating schedule: suppose the power plant has two units (indexed as unit 1 and unit 2) and each unit has two mills (indexed as mill 1 and mill 2 in each unit) and an overhaul action consumes five weeks, then the overhaul crew would firstly spend five weeks at mill 1 in unit 1, then five weeks at mill 1 in unit 2, then five weeks at mill 2 in unit 1, then five weeks at mill 2 in unit 2, and then the overhaul crew returns to mill 1 in unit 1 and repeats the whole process. Furthermore, the service crew is also assumed to operate on a similar rotating schedule such that the service opportunity is evenly distributed at both the unit level and the mill level, the same as the even distribution of the overhaul opportunity.

The rotating basis assumptions above are based on the following hypothesis: from the beginning of the non-contracted period, the maintenance resources would be relatively concentrated on the units/mills with relatively worse initial states, while the other units/mills with relatively better initial states receive less frequent maintenance; since all the units/mills are identical in terms of their physical structure in the power plant, such predisposed maintenance resources allocation would gradually bring the all the units/mills to a probabilistic equilibrium where each unit/mill is supposed to keep receiving nearly the same frequency of maintenance.

Based on such rotating schedule assumptions, in the long-term part of the non-contracted period, each mill is expected to wait for the same amounts of time until the corresponding maintenance crews return. As a result, we can consider the maintenance of a single generic mill which represents each of the $K \cdot N$ mills in the power plant; obviously, the generic mill does not exclusively occupy the maintenance resources in the power plant and instead it shares the maintenance crews with other mills based on the corresponding rotating schedules the same as any other mill in the power plant does. Focusing on a generic mill enables us to decompose the decision making problem from the power plant level to the mill level, which leads to a large scale of reduction in terms of the size of both the state space and action space in the ultimate mathematical model.

**Additional permutation rule**

Given such decomposition method, the modelling scope for maintenance decision making in the long-term would be focused on one individual generic mill, and therefore it is important to avoid the potential issue that one maintenance crew takes the generic mill offline for maintenance while at the same time the other maintenance crew takes another mill offline from the same unit for maintenance: as discussed in Section 4.1.1, maintenance engineers in the power plant developed a reasonable rule of thumb of not taking more than one mill offline in a unit for maintenance due to the concern of overly reducing the production capacity, based on the observed average electricity supply level in history. In our decomposition method, we shall further adopt such high-level insight that maintainers obtained in practice.

Hence we introduce an additional permutation rule here for the service rotating schedule, as part of the decomposition method: once the overhaul crew rotates to the unit which the generic mill belongs to (following the deterministic overhaul rotating schedule above) and visits a mill that is different from the generic mill, any service opportunity for the generic mill originally assigned by the deterministic service rotating schedule (as explained above) should be shifted immediately behind the overhaul time-window (which contains $OH$ time-steps). Such permutation rule assigns the priority to overhaul opportunities in a unit because the overhaul opportunity is much less frequent for a unit and overhaul improves the expected lifetime of a mill (which a service cannot achieve). Here is a numerical example of such permutation rule: suppose the overhaul crew rotates to a unit in every 20 weeks since its last arrival at the unit and would stay at the unit for 5 weeks before rotating to the next unit, then any service opportunity (assigned by the deterministic service rotating schedule) for the generic mill should be shifted immediately behind such 5 weeks overhaul time-window as long as the service time-window for the generic mill overlaps with the overhaul time-window for any other mill in the same unit.

Since the maintenance decision making in the non-contracted period is decomposed based on two different timescales (i.e. short-term and long-term), hence the name *two-timescale decomposition method.* The approximation effects of the decomposition method are mitigated by its rolling horizon nature as soon as the maintenance decision making enters the short-term part of the model.

**Adapting the two-timescale decomposition method for other case studies**

Note that the decomposition method is an optional modelling choice rather than a compulsory heuristic. The decomposition method enables one to reduce the size of both state space and action space in the part of the mathematical model which relates to the long-term maintenance decision making, which as a result reduces the total computational efforts required to solve the mathematical problem: such reduction effects can be very useful for case studies which involve large numbers of assets (see terminology in Section 4.1.3) such as our power plant case study, because (given the way the machine state is modelled in Chapter 2) the size of state space increases exponentially with the total number of assets and the size of action space increases linearly with the total number of assets in a company.

Such decomposition method however inevitably sacrifices the accuracy of the mathematical model, compared with the actual maintenance planning optimisation problem that is required to be solved in practice. We therefore recommend researchers and practitioners compare the total computational efforts required to solve the mathematical problem with and without such decomposition method in their own cases and benchmark against either the corresponding thresholds discussed in Section 2.4 or their own criteria, such that the best decision can be made on whether the decomposition method is necessary for their cases.

Additionally, the decomposition method is based on the assumption of even maintenance resources distribution in the power plant, and the even distribution assumption itself is based on the fact that the physical structure is identical at both the unit level and the mill level in the power plant. In other case studies a similar identical structure may not hold and researchers/practitioners need to accordingly adjust the deterministic rotating schedules of the maintenance crews for their cases, because the frequency of receiving maintenance resources varies between non-identical production systems/assets in the long term.

Furthermore, as part of the decomposition method, the permutation rule is based on the high-level insight (i.e. avoid taking more than one mill offline for maintenance in a unit) from the maintainers in the power plant and the insight itself is based on the observed electricity supply level contracted in history. Researchers/practitioners are recommended to adjust the maximum number of assets that can be taken offline from a production system based on the average production supply level contracted in history in their own cases.

Moreover, both the decomposition method and the permutation rule have room for improvement in our future studies: the decomposition method can further incorporate an extra time-window for emergency maintenance after every maintenance crew rotation cycle, and the time-length of such extra time-window can be determined based on the expected frequency of asset failures or the expected frequency of assets states deteriorating to certain undesirable levels. As for the permutation rule, it can further consider the expected occurrence frequency of the situations where more than one mill are offline already in a unit due to condition/performance failure and thus it is reasonable to perform maintenance on more than one mill simultaneously in a unit; meanwhile the rule can also be improved to consider the expected occurrence frequency of the situations where the overhaul rotates to a mill (different from the generic mill) in a unit but the mill is in a relatively good state and thus it does not require overhaul and therefore it is unnecessary to shift the service opportunity of the generic mill.

#### 4.2.1.2 Modelling framework

After introducing the decomposition method above which focuses on the non-contracted period, here we discuss the framework about how to model the complete optimal maintenance planning problem through integrating the decomposition method into the performance-centred maintenance approach.

We segment the entire maintenance planning period into three stages as shown in Figure 4.5: (1) the contracted period where operators may adjust the work-rate to mitigate/avoid the potential penalty cost; (2) the short-term part and (3) the long-term part of the non-contracted period where the work-rate is assumed to stay at a normal level (as explained in Section 2.2.1) and potential loss of sales can happen. We model the complete optimal maintenance planning problem by applying the MDP model from the performance-centred maintenance approach as follows: for Stage (1) and Stage (2), we shall scale up the MDP model to the plant level; for Stage (3) we shall modify the MDP model at the mill level, based on the decomposition method discussed above.

Since we segment the entire maintenance planning period into three stages, hence the name *three-stage hybrid MDP model.* Below we present the mathematical model in Section 4.2.2.

### 4.2.2 Mathematical model

The mathematical model for optimal decision making is comprised of three major components and two interface components, where the three major components correspond to the aforementioned three stages of the entire maintenance planning period and the two interface components aggregate the three major components together to capture the complete optimal maintenance planning problem.

We work backwards and introduce first the major component that corresponds to Stage (3) which involves only one representative mill. Next, we present the major component that corresponds to Stage (2) and then we provide the major component for Stage (1); note each such major component addresses all $K \cdot N$ mills at the plant level. Finally, we explain the two interface components. Note such backward introducing sequence for the major components aims at providing an easier understanding of the mathematical model for readers; we shall however solve the mathematical problem in a slightly different order and we shall specify such problem solving order and methods in Section 4.3.

We now introduce some mathematical notations for the entire maintenance planning period: the time-length of the entire maintenance planning period is denoted as $T$, and $T$ is infinite ( $T \to \infty$ ); the time-length of Stage (1), i.e. the contracted period, is denoted as $T_C$; the time-length of Stage (2), i.e. the short-term part of the non-contracted period, is denoted $T_S$; the time-length of Stage (3), i.e. the long-term part of the non-contracted period, is denoted $T_L$, and $T_L$ is infinite ($T_L \rightarrow \infty$). The relationship between the variables is illustrated in Figure 4.5.

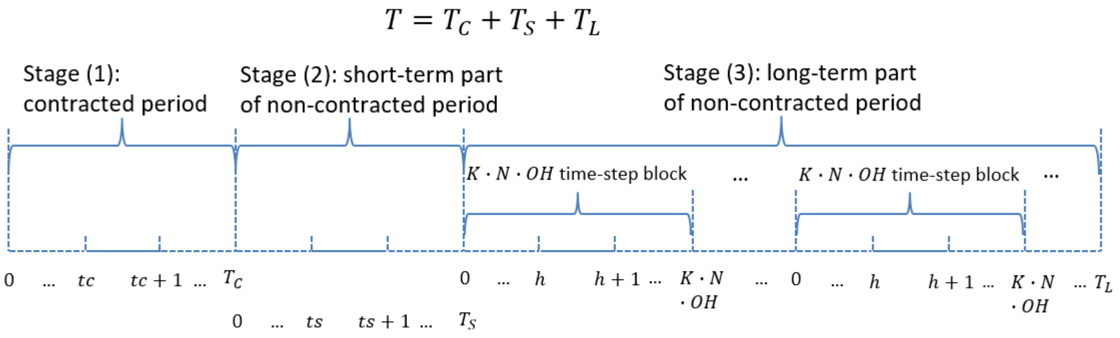

$T = T_C + T_S + T_L$
Stage (1): contracted period
Stage (2): short-term part of non-contracted period
Stage (3): long-term part of non-contracted period
$K \cdot N \cdot OH$ time-step block
...
$K \cdot N \cdot OH$ time-step block
...
0 ... $tc$ $tc+1$ ... $T_C$
0 ... $h$ $h+1$ ... $K \cdot N \cdot OH$
... 0 ... $h$ $h+1$ ... $K \cdot N \cdot OH$ ... $T_L$
0 ... $ts$ $ts+1$ ... $T_S$

Figure 4.5: Decomposition of infinite planning-horizon $T$

### 4.2.2.1 MDP model for Stage (3)

In the long-term part of the non-contracted period, i.e. Stage (3), we apply the two-stage decomposition method discussed in Section 4.2.1: we consider the maintenance of a single generic mill which represents each of the $K \cdot N$ mills in the power plant, and the maintenance crews follow the deterministic rotating schedules and permutation rule as discussed in Section 4.2.1.

The overhaul crew spends $OH$ time-steps at an mill before moving to the next one in the plant. Therefore in Stage (3) the generic mill on average needs to wait for $K \cdot N \cdot OH$ time-steps until the overhaul crew returns since its last arrival. As a result, we choose to further segment Stage (3) into infinite number of successive time blocks each of which contains $K \cdot N \cdot OH$ time-steps, as illustrated in Figure 4.5. The sequence of maintenance actions for the generic mill in every such time block is as follows: overhaul/service/no maintenance at the beginning of the time block, and service/ no maintenance at each following time-step in the time block. According to the deterministic service crew rotation schedule, the service opportunity occurs to the generic mill in every $K \cdot N$ time-steps and the opportunity would be further shifted based on the permutation rule.

Here we provide a numerical example for illustration: suppose the power plant contains four units and each unit consists of eight mills and an overhaul action consumes five weeks (i.e. $K = 4$, $N = 8$ and $OH = 5$); Stage (3) is segmented into successive time blocks each of which contains 160 weeks; the overhaul opportunity rotates to the generic mill at the fist week of every such time block, and the permuted service opportunity rotates to the generic mill at the following weeks in each such time block $\{1, 33, 66, 97, 129\}$; at the first week of each such time block, the maintainers need to choose between overhaul and service and no maintenance for the generic mill, and then the maintainers need to choose between service (if the service opportunity is available) and no maintenance for the generic mill at each remaining week after the chosen initial maintenance action in each time block.

The maintenance planning problem in Stage (3) for the generic mill can be summarised as follows: at the first time-step of every $K \cdot N \cdot OH$ time-step block, the maintainers need to choose between overhaul, service and no maintenance; then at each remaining time-step (in a time block) after the selected initial maintenance alternative, the maintainers need to choose between service (if the service opportunity occurs) and no maintenance. Below we model the optimal decision making by MDPs, based on adjusting the decomposed Bellman Equation (3.7) and Equation (3.8) from Section 3.3.

It is very important to note that for Stage (3) in the mathematical model, we only keep track of which time-step it is within a given $K \cdot N \cdot OH$ time-step block, as illustrated in Figure 4.5, rather than specifying which time-step it is since the beginning of Stage (3). In mathematical notation, let $h$ denote the time-step in any such given time block. Here we introduce some additional mathematical notations to facilitate the modelling work in Stage (3): let $\mathbf{x}_M$ denote the state of the generic mill and let $S_M$ denote the set of all potential mill states; let $H_{SM}$ denote the set of the time-steps within a $K \cdot N \cdot OH$ time-step block, where the maintenance alternatives include service; let $NM$ and $SM$ and $OM$ respectively denote no maintenance and service and overhaul; let $V_h^*(\mathbf{x}_M)$ denote the optimal value for mill state $\mathbf{x}_M \epsilon S_M$ at time-step $h$, and let $V_h^a(\mathbf{x}_M)$ denote the expected value of applying maintenance action $a$ to a given mill state $\mathbf{x}_M \epsilon S_M$ at time-step $h$ (see the definition of optimal value and expected value in Section 3.3.2 ). Below we provide the mathematical model for Stage (3):

For $1 \leq h \leq K\cdot N\cdot OH$, if $h = 1$ the optimal value $V_h^*(\mathbf{x}_M)$ for $\forall\, \mathbf{x}_M$ is defined in Equation (4.1)

$$V_h^*(\mathbf{x}_M) = \max_{a\epsilon\{NM,SM,OM\}} \{V_h^a(\mathbf{x}_M)\}, \,\text{for}\, h = 1, \tag{4.1}$$

and if $h > 1$ the optimal value $V_h^*(\mathbf{x}_M)$ for $\forall\, \mathbf{x}_M$ is defined in Equation (4.2)

$$V_h^*(\mathbf{x}_M) := \begin{cases} \max\limits_{a\epsilon\{NM,SM\}} \{V_h^a(\mathbf{x}_M)\}, & \text{for}\, h > 1 \,\text{and}\, h \in H_{SM}, \\ V_h^{a=NM}(\mathbf{x}_M), & \text{for}\, h > 1 \,\text{and}\, h \notin H_{SM}, \end{cases} \tag{4.2}$$

where the expected value $V_h^a(\mathbf{x}_M)$ in Equation (4.1)-(4.2) for $a = NM$ and $a = SM$ and $a = OM$ is defined in Equation (4.3)-(4.5) respectively

$$V_h^a(\mathbf{x}_M) := \begin{cases} E_{D_h}\{R_{NC}(\mathbf{x}_M, a, D_h)\} + \gamma \sum_{\mathbf{y}_M \epsilon S_M} P_{NC}(\mathbf{y}_M|\mathbf{x}_M, a) V_{h+1}^*(\mathbf{y}_M), \\ \text{for}\, a = NM, h < K\cdot N\cdot OH, \\ \\ E_{D_h}\{R_{NC}(\mathbf{x}_M, a, D_h)\} + \gamma \sum_{\mathbf{y}_M \epsilon S_M} P_{NC}(\mathbf{y}_M|\mathbf{x}_M, a) V_1^*(\mathbf{y}_M), \\ \text{for}\, a = NM, h = K\cdot N\cdot OH, \end{cases} \tag{4.3}$$

$$V_h^a(\mathbf{x}_M) := \begin{cases} C_{SM} + \gamma V_{h+1}^*(\mathbf{y}_M|\mathbf{x}_M, a), & \text{for}\, a = SM, h < K\cdot N\cdot OH, \\ C_{SM} + \gamma V_1^*(\mathbf{y}_M|\mathbf{x}_M, a), & \text{for}\, a = SM, h = K\cdot N\cdot OH, \end{cases} \tag{4.4}$$

$$V_h^a(\mathbf{x}_M) = C_{OM} + \gamma^{OH} V_{h+OH}^*(\mathbf{x}_M^{OM}), \,\text{for}\, a = OM, h = 1, \tag{4.5}$$

where in Equation (4.3) $D_h$ denotes the demand at time-step $h$; $R_{NC}$ denotes the reward function; $P_{NC}$ denotes the state transition probability; $V_1^*(\mathbf{y}_M)$ denotes the optimal value of mill state $\mathbf{y}_M$ at the first time-step of a $K\cdot N\cdot OH$ time-step block and $V_1^*(\mathbf{y}_M)$ is defined in Equation (4.1); in Equation (4.4) $C_{SM}$ denotes the fixed service maintenance cost; in Equation (4.5) $C_{OM}$ denotes the fixed overhaul maintenance cost, $\gamma^{OH}$ denotes the one time-step discount ratio $\gamma$ raised to the $OH - th$ power, and $\mathbf{x}_M^{OM}$ denotes the fixed mill state which a mill is restored to after overhaul.

For the MDP problem modelled above for Stage (3), the optimal maintenance alternative and the optimal value for any given mill state $\mathbf{x}_M$ and any time-step $h$ in any $K\cdot N\cdot OH$ time-step block are derived by solving Equation 4.1 or Equation 4.2, depending on whether $h = 1$ or $h > 1$ within the $K\cdot N\cdot OH$-time-step block.

#### 4.2.2.2 MDP model for Stage (2)

For the short-term part of the non-contracted period, i.e. Stage (2), we model the maintenance decision making problem at the plant level. Therefore here we explicitly consider the state of the entire plant. Here we highlight the difference of terminology between Stage (3) and Stage (2): in Stage (3) we use the term *maintenance alternative* where an alternative is either applying overhaul or service or no maintenance to the generic mill; in Stage (2) we instead use the following terms: *overhaul alternative* and *service alternative* where an alternative means applying overhaul or service respectively to a specific mill in the plant (for example apply service to the mill which is indexed as mill 4 in the unit which is indexed as unit 2 in the power plant, or initiate overhaul for mill 3 in unit 1 in the power plant). Note at a time-step in Stage (2), multiple overhaul alternatives and multiple service alternatives may exist.

The maintenance planning problem in Stage (2) can be summarised as follows: for each time-step in Stage (2), if the overhaul crew is idle, choose between

- applying no maintenance to any mill in the plant and

- applying a specific service alternative and
- *initiating* a specific overhaul alternative and
- applying a specific service alternative and *initiating* a specific overhaul alternative;

otherwise, if the overhaul crew is busy (i.e. already engaged in applying overhaul to a mill), choose between

- applying no *additional* maintenance to any mill in the plant and
- applying a specific service alternative.

### Notations in Stage (2)

Here we introduce some mathematical notations to facilitate the modelling work in Stage (2): let $\mathbf{x}_P$ denote the state of the plant, where $\mathbf{x}_P$ is a vector comprised of the states of all the units in the plant, and the state of a unit is furthermore a vector consisting of the states of all the mills in the unit; let $S_P$ denote the set of all potential plant states; let $ts$ denotes the time-step in Stage (2), and $ts \leq T_S$; let $T_{OM}$ to denote the last time-step in Stage (2) where overhaul can be initialised, and $T_{OM} = T_S - OH + 1$ which ensures the last potential overhaul planned in Stage (2) does not span into Stage (3); let $oh_{b,ts}$ denote how many time-steps an *undergoing* overhaul action $b$ has lasted *at the beginning* of time-step $ts$, and we shall abbreviate $oh_{b,ts}$ as $oh$ hereafter and $1 \leq oh < OH$; we introduce $oci_{ts} = 1/0$ to indicate the overhaul crew is idle/engaged respectively *at the beginning of* time-step $ts$ *before* any maintenance decision is made at that time step, and we shall abbreviate $oci_{ts}$ as $oci$ hereafter and $oci$ is initialised as 0 for Stage (2) and the value of $oci$ changes throughout Stage (2) depending on whether the overhaul crew is busy or idle; let $NAM$ denote *no additional maintenance* while overhaul is being performed, and let $NM$ denote *no maintenance*; let $a$ denote arbitrary *service alternative* and let $b$ denote arbitrary *overhaul alternative*; let $A_{SM}(\mathbf{x}_P)$ denote the set of all potential service alternatives for state $\mathbf{x}_P$, and let $A_{OM}(\mathbf{x}_P)$ denote the set of all potential overhaul alternatives for state

$\mathbf{x}_P$, and furthermore let $SM_b$ denote the service alternative which would conflict with given overhaul alternative $b$ (meaning both $SM_b$ and $b$ aim at maintaining the same mill), and let $A_{SM}(\mathbf{x}_P)/\{SM_b\}$ denote the set of all potential service alternatives for state $\mathbf{x}_P$ excluding the service alternative which conflicts with given overhaul alternative $b$, and hereafter we shall abbreviate $A_{SM}(\mathbf{x}_P)$ and $A_{OM}(\mathbf{x}_P)$ and $A_{SM}(\mathbf{x}_P)/\{SM_b\}$ respectively as $A_{SM}$ and $A_{OM}$ and $A_{SM}/\{SM_b\}$; let $V^*_{ts,oci=0}(\mathbf{x}_P)$ denote the optimal value for plant state $\mathbf{x}_P \epsilon S_P$ at time-step $ts$ for given overhaul crew status $oci = 0$ *at the beginning of* time-step $ts$, and let $V^*_{ts,oci=1}(\mathbf{x}_P, b, oh)$ denote the optimal value for plant state $\mathbf{x}_P \epsilon S_P$ at time-step $ts$ given that the undergoing overhaul alternative $b$ has lasted $oh$ time-steps *at the beginning of* time-step $ts$; let $V^a_{ts}(\mathbf{x}_P)$ and $V^b_{ts}(\mathbf{x}_P)$ and $V^{a,b}_{ts}(\mathbf{x}_P)$ respectively denote the expected value of (i) applying service alternative $a$ and (ii) *initiating* overhaul alternative $b$ and (iii) both applying service alternative $a$ and *initiating* overhaul alternative $b$ to a given plant state $\mathbf{x}_P$ at time-step $ts$, and let $V^a_{ts}(\mathbf{x}_P, b, oh)$ denote the expected value of applying service alternative $a$ to plant state $\mathbf{x}_P$ at time-step $ts$ given that the undergoing overhaul alternative $b$ has lasted $oh$ time-steps *at the beginning of* time-step $ts$.

**Mathematical model for Stage (2)**

For $ts \leq T_S$:

if $oci = 0$ and $ts \leq T_{OM}$ (meaning overhaul can be potentially initiated)

$$\begin{aligned} V^*_{ts,oci}(\mathbf{x}_P) = \max\{ & \max_{a \in \{NM, A_{SM}\}} \{V^a_{ts}(\mathbf{x}_P)\}, \max_{b \in A_{OM}} \{V^b_{ts}(\mathbf{x}_P)\}, \\ & \max_{a \in A_{SM}/\{SM_b\}, b \in A_{OM}} \{V^{a,b}_{ts}(\mathbf{x}_P)\}\}, \text{for}\, oci = 0, ts \leq T_{OM}, \end{aligned} \tag{4.6}$$

and if $oci = 0$ and $ts > T_{OM}$ (meaning no overhaul should be initiated)

$$V^*_{ts,oci}(\mathbf{x}_P) = \max_{a \in \{NM, A_{SM}\}} \{V^a_{ts}(\mathbf{x}_P)\}, \text{for}\, oci = 0, ts > T_{OM}, \tag{4.7}$$

and if $oci = 1$ (meaning the overhaul crew is engaged)

$$V^*_{ts,oci}(\mathbf{x}_P, b, oh) = \max_{a \in \{NAM, A_{SM}/\{SM_b\}\}} \{V^a_{ts}(\mathbf{x}_P, b, oh)\}, \text{ for } oci = 1, oh < OH, b \in A_{OM} \tag{4.8}$$

where the expected value $V^a_{ts}(\mathbf{x}_P)$ and $V^b_{ts}(\mathbf{x}_P)$ and $V^{a,b}_{ts}(\mathbf{x}_P)$ and $V^a_{ts}(\mathbf{x}_P, b, oh)$ in Equation (4.6)-(4.8) for $a \in \{NM, A_{SM}\}$ and $b \in A_{OM}$ and $a \in A_{SM}/\{SM_b\}, b \in A_{OM}$ and $b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}$ is defined in Equation (4.9)-(4.12) respectively

$$V^a_{ts}(\mathbf{x}_P) = E_{\boldsymbol{D}_{ts}}\{R_{NC}(\mathbf{x}_P, a, \boldsymbol{D}_{ts})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_{NC}(\mathbf{y}_P|\mathbf{x}_P, a) V^*_{ts+1,oci=0}(\mathbf{y}_P),$$
$$\text{for } a \in \{NM, A_{SM}\}, \tag{4.9}$$

$$V^b_{ts}(\mathbf{x}_P) = E_{\boldsymbol{D}_{ts}}\{R_{NC}(\mathbf{x}_P, b, \boldsymbol{D}_{ts})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_{NC}(\mathbf{y}_P|\mathbf{x}_P, b) V^*_{ts+1,oci=1}(\mathbf{y}_P, b, oh = 1),$$
$$\text{for } b \in A_{OM}, \tag{4.10}$$

$$V^{a,b}_{ts}(\mathbf{x}_P) = E_{\boldsymbol{D}_{ts}}\{R_{NC}(\mathbf{x}_P, a, b, \boldsymbol{D}_{ts})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_{NC}(\mathbf{y}_P|\mathbf{x}_P, a, b)$$
$$V^*_{ts+1,oci=1}(\mathbf{y}_P, b, oh = 1), \text{ for } a \in A_{SM}/\{SM_b\}, b \in A_{OM}, \tag{4.11}$$

$$V_{ts}^{a}(\mathbf{x}_P, b, oh) = \begin{cases} E_{\boldsymbol{D}_{ts}}\{R_{NC}(\mathbf{x}_P, a, b, \boldsymbol{D}_{ts})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_{NC}(\mathbf{y}_P | \mathbf{x}_P, a, b, oh) \\ V_{ts+1,oci=1}^{*}(\mathbf{y}_P, b, oh+1), \\ \text{for } 1 \leq oh < OH-1, b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}, \\ \\ E_{\boldsymbol{D}_{ts}}\{R_{NC}(\mathbf{x}_P, a, b, \boldsymbol{D}_{ts})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_{NC}(\mathbf{y}_P | \mathbf{x}_P, a, b, oh) \\ V_{ts+1,oci=0}^{*}(\mathbf{y}_P), \\ \text{for } oh = OH-1, b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}. \end{cases} \tag{4.12}$$

where in Equation (4.9)-(4.12) $\boldsymbol{D}_{ts}$ is a vector that consists of the output demand for each unit in the power plant at time-step $ts$; $R_{NC}$ denotes the reward function; $P_{NC}$ denotes the state transition probability for the plant state and Section 2.2.3 specifies how to derive such transition probabilities; in Equation (4.12) $oh = OH - 1$ means the undergoing overhaul action $b$ has lasted $OH - 1$ time-steps *at the beginning* of time-step $ts$, and such overhaul action will be finished *at the end* of time-step $ts$ and hence the overhaul crew will be idle *at the beginning* of the next time-step (hence $oci = 0$ for $ts + 1$).

For the MDP problem modelled above for Stage (2), the optimal maintenance decision and the optimal value for any given plant state $\mathbf{x}_P$ and any time-step $ts$ are derived by solving Equation (4.6) or Equation (4.7) or Equation (4.8), depending on the value of $oci$ (i.e. whether $oci = 0$ or $oci = 1$ ) and $ts$ (i.e. whether $ts \leq T_{OM}$ or $ts > T_{OM}$).

#### 4.2.2.3 MDP model for Stage (1)

In this section, we focus on modelling the optimal decision making in the contracted period, i.e. Stage (1), of the maintenance planning problem. Given the discussion in Section 4.2.1 regarding the mathematical modelling framework for the entire maintenance planning problem, it is easy to see that the modelling work for Stage (1) is very similar to the modelling work in Stage (2), and the main difference include: in Stage (1) the mathematical model should capture that the operators would potentially adjust the work-rate of the machines in order to meet the contracted supply; in Stage (1) penalty cost occurs if the contracted supply is not fulfilled. Here we shall reuse the majority of the mathematical notations and mathematical model from Section 4.2.2.2, and the new notations introduced only for Stage (1) are given below.

We introduce $tc$ to denote the time-step since the beginning of Stage (1), and $tc \leq TC$; we introduce $R_C$ to denote the reward function and introduce $P_C$ to denote the plant state transition probability in Stage (1), and compared with their counterparts in Stage (2) (i.e. $R_{NC}$ and $P_{NC}$ respectively) they furthermore depend on the work-rate level of each unit in the plant, and additionally here $R_C$ describes penalty cost for unfulfilled contracted supply; we introduce $\boldsymbol{wr}_{tc}$ to denote the work-rate levels in the power plant, where $\boldsymbol{wr}_{tc}$ is a vector consists of the work-rate of each unit in the power plant at time-step $tc$.

We reuse $T_{OM}$ from Section 4.2.2.2 to denote the last time-step in Stage (1) where overhaul can be initialised, but for Stage (1) we update the value of $T_{OM}$ as $T_{OM} = T_C - OH + 1$ which ensures the last potential overhaul planned in Stage (1) does not span into Stage (2). Regarding the other notations that we reuse from Section 4.2.2.2, their domain meaning is simply updated from the context of Stage (2) to the context of Stage (1) and we shall not further explain.

**Mathematical model for Stage (1)**

For $tc \leq T_C$:

if $oci = 0$ and $tc \leq T_{OM}$ (meaning overhaul can be potentially initiated)

$$\begin{aligned} V^*_{tc,oci}(\mathbf{x}_P) = \max\{ & \max_{a \in \{NM, A_{SM}\}} \{V^a_{tc}(\mathbf{x}_P)\}, \max_{b \in A_{OM}} \{V^b_{tc}(\mathbf{x}_P)\}, \\ & \max_{a \in A_{SM}/\{SM_b\}, b \in A_{OM}} \{V^{a,b}_{tc}(\mathbf{x}_P)\}\}, \text{ for } oci = 0, tc \leq T_{OM}, \end{aligned} \tag{4.13}$$

and if $oci = 0$ and $tc > T_{OM}$ (meaning no overhaul should be initiated)

$$V^*_{tc,oci}(\mathbf{x}_P) = \max_{a\in\{NM,A_{SM}\}} \{V^a_{tc}(\mathbf{x}_P)\}, \,\text{for}\, oci = 0, tc > T_{OM}, \tag{4.14}$$

and if $oci = 1$ (meaning the overhaul crew is engaged)

$$V^*_{tc,oci}(\mathbf{x}_P, b, oh) = \max_{a\in\{NAM,A_{SM}/\{SM_b\}\}} \{V^a_{tc}(\mathbf{x}_P, b, oh)\}, \,\text{for}\, oci = 1, oh < OH, b \in A_{OM} \tag{4.15}$$

where the expected value $V^a_{tc}(\mathbf{x}_P)$ and $V^b_{tc}(\mathbf{x}_P)$ and $V^{a,b}_{tc}(\mathbf{x}_P)$ and $V^a_{tc}(\mathbf{x}_P, b, oh)$ in Equation (4.13)-(4.15) for $a \in \{NM, A_{SM}\}$ and $b \in A_{OM}$ and $a \in A_{SM}/\{SM_b\}, b \in A_{OM}$ and $b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}$ is defined in Equation (4.16)-(4.19) respectively

$$V^a_{tc}(\mathbf{x}_P) = E_{\boldsymbol{D}_{tc}}\{R_C(\mathbf{x}_P, a, \boldsymbol{D}_{tc}, \boldsymbol{wr}_{tc})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_C(\mathbf{y}_P|\mathbf{x}_P, a, \boldsymbol{wr}_{tc}) V^*_{tc+1,oci=0}(\mathbf{y}_P), \\ \text{for}\, a \in \{NM, A_{SM}\}, \tag{4.16}$$

$$V^b_{tc}(\mathbf{x}_P) = E_{\boldsymbol{D}_{tc}}\{R_C(\mathbf{x}_P, b, \boldsymbol{D}_{tc}, \boldsymbol{wr}_{tc})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_C(\mathbf{y}_P|\mathbf{x}_P, b, \boldsymbol{wr}_{tc}) \\ V^*_{tc+1,oci=1}(\mathbf{y}_P, b, oh = 1), \,\text{for}\, b \in A_{OM}, \tag{4.17}$$

$$V^{a,b}_{tc}(\mathbf{x}_P) = E_{\boldsymbol{D}_{tc}}\{R_C(\mathbf{x}_P, a, b, \boldsymbol{D}_{tc}, \boldsymbol{wr}_{tc})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_C(\mathbf{y}_P|\mathbf{x}_P, a, b, \boldsymbol{wr}_{tc}) \\ V^*_{tc+1,oci=1}(\mathbf{y}_P, b, oh = 1), \,\text{for}\, a \in A_{SM}/\{SM_b\}, b \in A_{OM}, \tag{4.18}$$

$$V_{tc}^{a}(\mathbf{x}_P, b, oh) = \begin{cases} E_{\boldsymbol{D}_{tc}}\{R_C(\mathbf{x}_P, a, b, \boldsymbol{D}_{tc}, \boldsymbol{wr}_{tc})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_C(\mathbf{y}_P | \mathbf{x}_P, a, b, oh, \boldsymbol{wr}_{tc}) \\ V_{tc+1,oci=1}^{*}(\mathbf{y}_P, b, oh+1), \\ \text{for}\, 1 \leq oh < OH - 1, b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}, \\ \\ E_{\boldsymbol{D}_{tc}}\{R_C(\mathbf{x}_P, a, b, \boldsymbol{D}_{tc}, \boldsymbol{wr}_{tc})\} + \gamma \sum_{\mathbf{y}_P \epsilon S_P} P_C(\mathbf{y}_P | \mathbf{x}_P, a, b, oh, \boldsymbol{wr}_{tc}) \\ V_{tc+1,oci=0}^{*}(\mathbf{y}_P), \\ \text{for}\, oh = OH - 1, b \in A_{OM}, a \in \{NAM, A_{SM}/\{SM_b\}\}. \end{cases} \tag{4.19}$$

Regarding the MDP problem modelled above for Stage (1), the optimal maintenance decision and the optimal value for any given plant state $\mathbf{x}_P$ and any time-step $tc$ are derived by solving Equation (4.13) or Equation (4.14) or Equation (4.15), depending on the value of $oci$ (i.e. whether $oci = 0$ or $oci = 1$ ) and $tc$ (i.e. whether $tc \leq T_{OM}$ or $tc > T_{OM}$).

#### 4.2.2.4 Interfaces between different stages

So far we separately modelled the optimal maintenance decision making in each of the three stages of the maintenance planning problem, in Section 4.2.2.1-4.2.2.3 respectively. Here we shall introduce the last components of the mathematical model: interfaces which aggregate the three major components developed in Section 4.2.2.1-4.2.2.3 such that the ultimate mathematical model can capture the impact of the decision making in a stage on the later stage(s).

**Interface between Stage (2) and Stage (3)**

The maintenance decision and the plant state at the last time-step in Stage (2), i.e. time-step $T_S$, impact what state the plant evolves into at the next time-step and hence influence the optimal value at that time-step, i.e. the first time-step in Stage (3). We therefore define Equation (4.20) to capture the influence from Stage (2) to Stage (3):

$$V^*_{T_S+1,oci=0}(\mathbf{x}_P) = \sum_{k=1}^{K}\sum_{n=1}^{N} V^*_{h=1}(\mathbf{x}_M = \mathbf{x}_{k,n} \mid \mathbf{x}_P), \forall\, \mathbf{x}_P \tag{4.20}$$

where $\mathbf{x}_{k,n}$ denotes the mill state of the mill which is indexed as mill $n$ in the unit which is indexed as unit $k$ in the power plant.

**Interface between Stage (1) and Stage (2)**

Similarly, we define Equation (4.21) to capture the influence from Stage (1) to Stage (2)

$$V^*_{T_C+1,oci=0}(\mathbf{x}_P) = V^*_{ts=1,oci=0}(\mathbf{x}_P), \forall\, \mathbf{x}_P. \tag{4.21}$$

With the two interface components developed, the entire mathematical model we built above in Section 4.2.2 can approximately capture the impact of the decision making at an earlier time-step on the decision making at every later time-steps up to the horizon of the entire maintenance planning period. In other words, the three-stage hybrid MDP model supports the optimal decision making for the complete maintenance planning problem of the power plant case study.

The three-stage hybrid MDP model is built in the context of the power plant case; such model can be relatively easily adapted for other cases which share the same hierarchical structure (the structure is specified in Section 4.3); see discussion regarding model generalisation in Section 4.2.1.

#### 4.2.2.5 Inputs and outputs of the model

Here we further clarify (1) what input parameter values should be specified by practitioners when applying the three-stage hybrid MDP model and (2) what output results can be expected after solving the mathematical problem.

As shown in Table 4.8, the PCM approach does not require impractical specifications of input parameters at the plant level; instead the elicitation of complex input parameters is decomposed to the unit or mill level. Note that the formula expression of certain input parameters (these are $wr$ and $\boldsymbol{D}_t$ in Table 4.8) are case-based (as shown in the case studies discussed in Section 4.1.4 and Chapter 5), and hence here we choose not to further discuss their specific formula expressions.

Regarding the computation results of solving the three-stage hybrid MDP problem, as shown in Table 4.9, given an operators' decision regime, practitioners can derive an estimate for the expected value of applying any maintenance action to the initial plant state; by comparing such estimated expected values between different maintenance actions, practitioners can then derive an estimate for the optimal maintenance action (to be applied to the initial plant state) and an estimate for the associated optimal value. In mathematical expression, the estimated optimal maintenance action is $\arg\max_{a \in A(X_P)} \bar{Q}_{wr}(X_P, a, tc = \mathit{1})$ and the estimated associated optimal value is $\max_{a \in A(X_P)} \bar{Q}_{wr}(X_P, a, tc = \mathit{1})$. Additionally practitioners can modify the input parameter (that is $wr$) for different operators' decision regimes and compare the aforementioned estimated optimal values to evaluate which operators' decision regime is approximately the optimal choice. In mathematical expression, the estimated optimal operators' decision regime is the one of which the estimated optimal value is equal to $\max_{wr}(\max_{a \in A(X_P)} \bar{Q}_{wr}(X_P, a, tc = \mathit{1}))$. In summary, regarding the output results, practitioners can (1) approximately identify the optimal operators' decision regime from available choices and also (2) estimate the optimal maintenance action and the associated optimal value for the given initial plant state.

From the perspective of practitioners, the logic flow of applying the PCM approach to facilitate their decision making can be summarised as Figure 4.6: (1) specify the input parameters listed in Table 4.8, (2) implement the three-stage hybrid MDP model (for example implementing in MATLAB), (3) apply certain heuristics (for example the heuristics developed in Section 4.3) to solve the mathematical problem and (4) make maintenance and production decisions based on the computation results as discussed above.

| Input parameter | Meaning |
|---|---|
| $T$ | The maintenance planning period |
| $T_L$ | The long-term part of the non-contracted period |
| $T_s$ | The short-term part of the non-contracted period |
| $T_C$ | The contracted period |
| $K$ | The amount of units in the power plant |
| $N$ | The amount of mills per unit |
| $H_{SM}$ | The set of the time-steps where the maintenance alternatives include service in Stage (3) of the model; specified based on the deterministic rotating schedules and permutation rule in Section 4.2.1.1. |
| $wr$ | The work-rate function that applies to any unit at any time-step in Stage (1) of the model for a given operators' decision regime |
| $\boldsymbol{D}_t$ | The vector that consists of the demand (distribution) at time-step $\forall t \leq T$ for unit $(1, 2, ..., K)$ |
| $S_M$ | The set of all potential mill states |
| $A_M$ | The set of all potential maintenance actions at the individual mill level |
| $OH$ | The amount of time-steps an overhaul action consumes |
| $SH$ | The amount of time-steps a service action consumes |
| $C_{SM}$ | The service maintenance cost |
| $C_{OM}$ | The overhaul maintenance cost |
| $R_S$ | The sales price per unit of supply |
| $R_P$ | The penalty cost per unit of unfulfilled contracted supply in Stage (1) of the model |
| $P_M$ | Mill state transition matrix |
| $\gamma$ | One time-step discount ratio |
| $X_P$ for $tc = 1$ | The initial plant state |

Table 4.8: Inputs for three-stage hybrid MDP model

| Computation results | Meaning |
|---|---|
| $\bar{Q}_{wr}(X_P, a, tc = 1)$ for $\forall\, a \in A(X_P)$ given $(X_P, wr)$ | The estimate of expected value for applying any maintenance action $a$ to the given initial plant state $X_P$ given a specified work-rate function $wr$ |

Table 4.9: Computation results of solving three-stage hybrid MDP problem

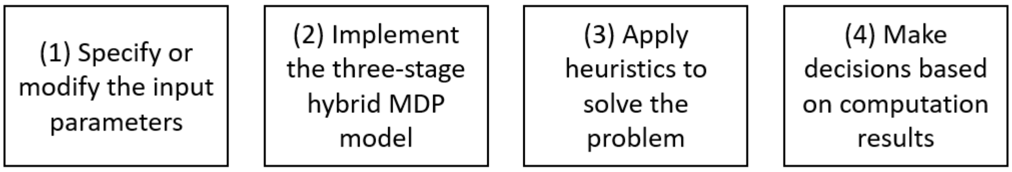

(1) Specify or modify the input parameters
(2) Implement the three-stage hybrid MDP model
(3) Apply heuristics to solve the problem
(4) Make decisions based on computation results

Figure 4.6: Logic flow of modelling and implementation

## 4.3 Heuristics for the three-stage hybrid MDP problem

In order to solve the three-stage hybrid MDP problem for the case study, we shall apply a set of (heuristic) methods to the MDP problem in a certain sequence as specified below.

We shall first apply the *value iteration* computation method (see Section 3.2) to the small size MDP problem arising from Stage (3) of the case study (see the MDP model in Section 4.2.2.1) and store the computational results in brute-force lookup tables. Such exact computational results are fed to the large-size MDP models for earlier stages (i.e. Stage (1) and Stage (2)) of the the case study via Equation (4.20) in Section 4.2.2.4. Then we shall aggregate the large-size MDP problems arising from Stage (1) and Stage (2) of the case study via Equation (4.21) into a single MDP problem, and apply Q-learning (see Section 3.3) to the aggregated single MDP problem and additionally we shall use the *polynomial function method* (see Section 3.4) to store the computational results for such aggregated problem. Figure 4.7 illustrates the application scopes of such methods.

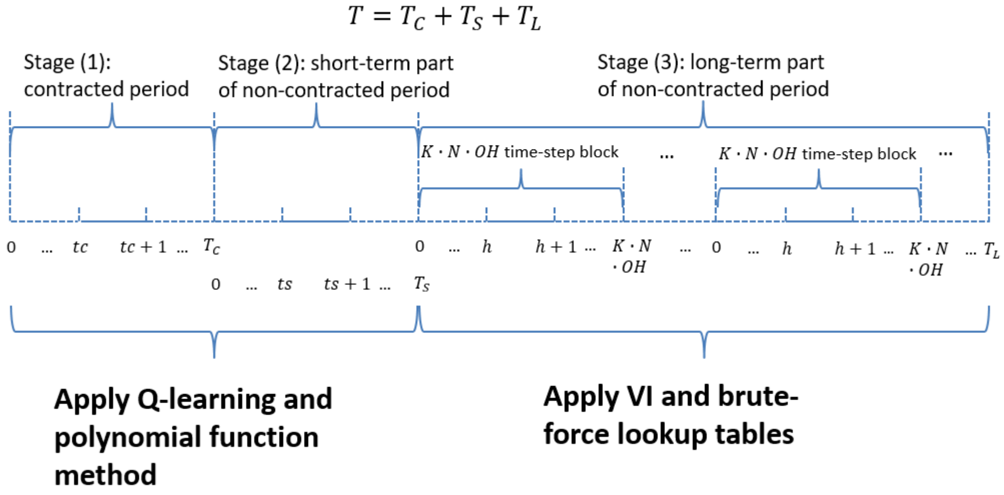

$T = T_C + T_S + T_L$
Stage (1): contracted period
Stage (2): short-term part of non-contracted period
Stage (3): long-term part of non-contracted period
$K \cdot N \cdot OH$ time-step block
...
$K \cdot N \cdot OH$ time-step block
...
0 ... $tc$ $tc+1$ ... $T_C$
0 ... $ts$ $ts+1$ ... $T_S$
0 ... $h$ $h+1$ ... $K \cdot N \cdot OH$ ... 0 ... $h$ $h+1$ ... $K \cdot N \cdot OH$ ... $T_L$
Apply Q-learning and polynomial function method
Apply VI and brute-force lookup tables

Figure 4.7: Methods application scopes

By applying the set of methods in such sequence to the three-stage hybrid MDP model developed in Section 4.2, for arbitrarily given initial state of the three-stage hybrid MDP problem we hope to obtain a near-optimal policy which prescribes a near-optimal action for each state the hybrid MDP model would potentially evolve into at each time-step during the entire planning period; additionally we hope the computational time and data-storage cost and ultimate data accuracy level are acceptable in practice for the three-stage hybrid MDP problem (see discussion of corresponding benchmark thresholds in Section 2.3). We shall discuss the corresponding numerical test results in Chapter 5.

Regarding the value iteration and Q-learning computation methods, we have specified them in Section 3.2 and Section 3.3 respectively, and their application to the three-stage hybrid MDP problem is relatively straightforward. Hence we shall not extend the discussion here. Instead, in Section 4.3.1 we shall justify the choice of polynomial function method for the aggregated MDP problem of Stage (1) and Stage (2) in the case study and also specify how to apply such method to the power plant case study, and then we shall generalise the context and extend the discussion to other case studies of interest; in Section 4.3.2 we shall specify how such data-storage heuristic interacts with Q-learning.

### 4.3.1 Polynomial function method

The essence of the applying the polynomial function method to the power plant case study is approximating the numerical relationship between the plant state and the expected value in the aggregated MDP problem by polynomial functions. More specifically speaking, for each stage (i.e. Stage (1) or Stage (2)) in the aggregated MDP problem, a separate polynomial function is defined to approximate such numerical relationship for each combination of maintenance choice and time-step in the MDP problem. Each such polynomial function contains a certain number of free parameters to be tuned; by *tuning* we mean adapting the values of the free parameters such that the polynomial function can yield acceptable value approximation quality in practice.

In this section we shall first focus the context to the power plant case study and (i) justify the choice of the polynomial function method and (ii) specify the degree and independent variables of the polynomial functions and (iii) introduce a method to reduce the number of free parameters in such functions and (iv) define such functions, and then we shall generalise the context and (v) discuss whether and how to apply the polynomial function method to other case studies of interest.

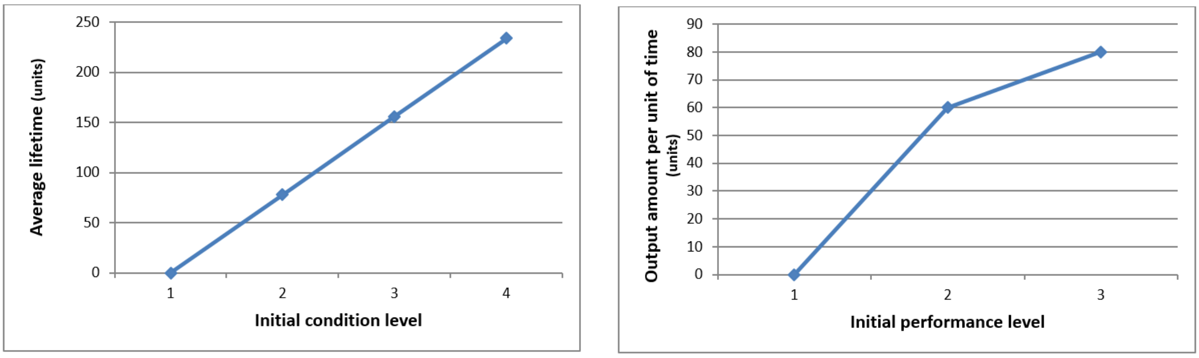
**Justification for choosing the polynomial function method**

The polynomial function method is chosen based on exploiting the background knowledge of the case study. In the hybrid MDP model of the case study, the plant state is a vector comprised of the degradation condition and operational performance of each mill in the power plant, and such condition/performance impacts the expected profit (i.e. the expected value modelled in the hybrid MDP model) by determining the lifetime/output-rate of the corresponding mill (see the discussion about the case study background in Section 4.1). Furthermore, the numerical test below shows the numerical relationship between the condition/performance and such average lifetime/output-rate follows concave patterns in the case study, which supports the modelling choice of approximating the numerical relationship of interest by polynomial functions defined on the condition and performance of each mill in the power plant.

The numerical test aforementioned is as follows: change the initial condition and initial performance of a representative mill each from the lowest possible level to the highest possible level, and accordingly check the values of two measures: (1) the average lifetime of the mill and (2) the amount of output the mill contributes per unit of time; then extrapolate such numerical relationships from the discrete condition and performance measurement scales to continuous scales. The results are shown in Figure 4.8.

**Independent variables and function degree**

The independent variables in each such polynomial function are the conditions and performance of all mills in the power plant: $\mathbf{c}_{k,n}$ and $\mathbf{p}_{k,n}$ for $\forall\,(k,n)$, where $\mathbf{c}_{k,n}$ and $\mathbf{p}_{k,n}$ denote the condition and performance respectively for mill $n$ in unit $k$ (where $n \leq N$ and $k \leq K$; again, $K$ denotes the total number of units in the power plant, and $N$ denotes



(a) Average lifetime

(b) Output amount per time unit

Figure 4.8: Data patterns (normal work-rate level)

the number of mills per unit).

The trade-off in deciding the degree of the polynomial functions is as follows: if the degree is too low, the value approximation quality would be undesirable; if the degree is too high, the number of free parameters would be relatively large and the computational cost of tuning them may be impractically high (see discussion on benchmark in Section 2.3). In order to find a balance between the value approximation quality and the total number of free parameters, we experimented from the minimum plausible function degree (which is 2, given the concave data patterns aforementioned) and it already derives relatively accurate approximation (we shall specify the numerical results in Chapter 5); we therefore choose to use *quadratic functions* to approximate the numerical relationship for the power plant case study.

Furthermore, since each unit in the plant has its separate supply contract and the units cannot directly exploit each other's potential extra production capacity (as discussed in Section 4.1.2), we additionally choose to define each such quadratic function of interest as the sum of multiple *smaller* size quadratic functions each of which only involves $\mathbf{c}_{k,n}$ and $\mathbf{p}_{k,n}$ from a single unit. As a numerical example, for $K = 2$ and $N = 2$, such *smaller* size quadratic function which only involves unit 1 takes the form $\sum_{n=1}^{N} w(\mathbf{c}_{1,n}^2)\cdot \mathbf{c}_{1,n}^2 + \sum_{n=1}^{N} w(\mathbf{p}_{1,n}^2)\cdot \mathbf{p}_{1,n}^2 + w(\mathbf{c}_{1,1}, \mathbf{c}_{1,2})\cdot \mathbf{c}_{1,1}\cdot \mathbf{c}_{1,2} + \sum_{e=1}^{N}\sum_{f=1}^{N} w(\mathbf{c}_{1,e}, \mathbf{p}_{1,f})\cdot \mathbf{c}_{1,e}\cdot \mathbf{p}_{1,f} + w(\mathbf{p}_{1,1}, \mathbf{p}_{1,2})\cdot \mathbf{p}_{1,1}\cdot \mathbf{p}_{1,2} + \sum_{n=1}^{N} w(\mathbf{c}_{1,n})\cdot \mathbf{c}_{1,n} +$

$\sum_{n=1}^{N} w(\mathbf{p}_{1,n})\cdot \mathbf{p}_{1,n} + w(0)$, where $w(\bullet)$ denotes free parameters; the other *smaller* quadratic function only involves unit 2, and the (full size) quadratic function of interest is the sum of such two *smaller* quadratic functions.

**Reducing the number of free parameters**

Hereafter in this section, by quadratic functions we mean the aforementioned full size quadratic functions, unless stated otherwise. Using quadratic functions to approximately store the computational results obviously mitigates the data-storage burden, compared with the brute-force lookup tables. However, such modelling choice in general may induce a large number of free parameters to be tuned. More specifically speaking, in the context of the power plant case, the number of free parameters in each quadratic function grows exponentially with the number of mills per unit in the power plant, mostly due to the exponential increment of the number of cross-products in the polynomial functions: the total number of the cross products is $K(2N^2 - N)$ for each quadratic function, and the total number of free parameters for all the terms (including all cross-products and all the other terms) in a quadratic function would be $K(2N^2 + 3N + 1)$. Hence a relatively small number of mills per unit can lead to a relatively large number of free parameters.

Therefore we develop a method which reduces the total amount of free parameters to a fixed and relatively small number for each quadratic function regardless of the number of the mills per unit in the power plant, *without* sacrificing the value approximation quality of the quadratic functions; we refer to such method as the *parameters number bounding method.* Generally speaking, for each quadratic function, such method requires evaluating the homogeneity of the terms in the function based on the case study background knowledge, and then the method requires combining the homogeneous terms such that they share the same free parameter; from a mathematical viewpoint, such arrangement does not induce extra value approximation inaccuracy for the quadratic functions.

More specifically speaking, according to the case study background knowledge, the mills have identical physical structure in the power plant; therefore from the perspective of the quadratic functions, $\mathbf{c}_{k,1}, ..., \mathbf{c}_{k,N}$ of the mills from the same unit are equally important variables regardless of which mill they belong to, and $\mathbf{p}_{k,1}, ..., \mathbf{p}_{k,N}$ of the mills from the same unit are also equally important variables. Based on such homogeneous property of independent variables, we can determine whether two arbitrary terms in a quadratic function are homogeneous, and we shall combine all homogeneous terms together to share one parameters. As a numeral example, $\mathbf{c}_{1,1}\cdot \mathbf{p}_{1,2}$ and $\mathbf{c}_{1,3}\cdot \mathbf{p}_{1,4}$ are homogeneous terms as they are both a cross-product of the condition and performance from two different mills in unit 1, and as a result we can combine $\mathbf{c}_{1,1}\cdot \mathbf{p}_{1,2}$ and $\mathbf{c}_{1,3}\cdot \mathbf{p}_{1,4}$ to share one free parameter instead of assigning a separate free parameter to each individual term; in fact all terms in the form of $\mathbf{c}_{1,e}\cdot \mathbf{p}_{1,f}$ are considered homogeneous and thus they shall all be combined to share just one free parameter regardless of how many mills exist in unit 1 ($\forall e, f \leq N, e \neq f$ ). We provide the full list of homogeneous terms in Table 4.10 for the quadratic functions.

As a result, by applying the *parameters number bounding method*, each quadratic function would contain $9 * K$ free parameters (as readers shall see immediately below), rather than $K(2N^2 + 3N + 1)$ free parameters.

| **Different forms of homogeneous terms** | **Meaning (in the same unit)** |
|---|---|
| $\mathbf{c}_{k,e}\mathbf{c}_{k,f}$ $(\forall k \leq K, \forall e, f \leq N, e \neq f)$ | cross-product of the conditions from two arbitrary different mills |
| $\mathbf{c}^2_{k,e}(\forall k \leq K, \forall e \leq N)$ | square of the condition of an arbitrary mill |
| $\mathbf{p}_{k,e}\mathbf{p}_{k,f}(\forall k \leq K, \forall e, f \leq N, e \neq f)$ | cross-product of the performance from two arbitrary different mills |
| $\mathbf{p}^2_{k,e}(\forall k \leq K, \forall e \leq N)$ | square of the performance of an arbitrary mill |
| $\mathbf{c}_{k,e}\mathbf{p}_{k,f}(\forall k \leq K, \forall e, f \leq N, e \neq f)$ | cross-product of the condition and performance from two arbitrary different mills |
| $\mathbf{c}_{k,e}\mathbf{p}_{k,e}(\forall k \leq K, \forall e \leq N)$ | cross-product of the condition and performance of an arbitrary mill |
| $\mathbf{c}_{k,e}(\forall k \leq K, \forall e \leq N)$ | condition of an arbitrary mill |
| $\mathbf{p}_{k,e}(\forall k \leq K, \forall e \leq N)$ | performance of an arbitrary mill |

Table 4.10: Homogeneous terms in quadratic functions

**Defining the quadratic functions**

Here we shall define the quadratic functions for the power plant case study. As discussed above, for Stage (1) and Stage (2), a separate polynomial function is defined to approximate the numerical relationship of interest for each combination of maintenance choice and time-step in the MDP problem.

In terms of the potential maintenance choices in Stage (1) and Stage (2), they are already discussed in Section 4.2.2.2; we shall group such choices into three categories below and then define the quadratic functions for each such category rather than for each specific maintenance choice. Note that no value approximation accuracy would be sacrificed in such arrangement. By using the same mathematical notations from Section 4.2.2.2 and Section 4.2.2.3, below we first define the quadratic functions of interest for Stage (2) and then Stage (1) of the three-stage hybrid MDP model.

For Stage (2):

(1) if the maintenance choice at time-step $ts$ is either (i) applying no maintenance to any mill in the plant or (ii) applying a specific service alternative, then

$$\begin{aligned}\overline{Q}(\mathbf{x}_P, a, ts) = &\sum_{k=1}^{K}\{w_{cc}^{(2)}(a,ts,k)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{c}_{k,e}\mathbf{c}_{k,f} \\ &+ w_{c2}^{(2)}(a,ts,k)\sum_{e=1}^{N}\mathbf{c}_{k,e}^2 + w_{pp}^{(2)}(a,ts,k)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{p}_{k,e}\mathbf{p}_{k,f} \\ &+ w_{p2}^{(2)}(a,ts,k)\sum_{e=1}^{N}\mathbf{p}_{k,e}^2 + w_{cp}^{(2)}(a,ts,k)\sum_{e=1}^{N}\sum_{f=1,f\neq e}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,f} \\ &+ w_{pc}^{(2)}(a,ts,k)\sum_{e=1}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,e} + w_{c}^{(1)}(a,ts,k)\sum_{e=1}^{N}\mathbf{c}_{k,e} \\ &+ w_{p}^{(1)}(a,ts,k)\sum_{e=1}^{N}\mathbf{p}_{k,e} + w^{(0)}(a,ts,k)\},\ \forall\, a\epsilon\{NM, A_{SM}\} \qquad (4.22)\end{aligned}$$

where $\overline{Q}$ is the output value of the quadratic function and $\overline{Q}(\mathbf{x}_P, a, ts)$ is the estimate of targeted expected value $V_{ts}^{a}(\mathbf{x}_P)$ (see Section 4.2.2.2) for $\forall\,(\mathbf{x}_P, a, ts)$; the plant state is defined as $\mathbf{x}_P := (\mathbf{c}_{1,1}, \mathbf{p}_{1,1}, ..., \mathbf{c}_{K,N}, \mathbf{p}_{K,N})$;

(2) if the maintenance choice at time-step $ts$ is either (i) *initiating* a specific overhaul alternative or (ii) continuing with the undergoing overhaul action, then

$$\begin{aligned}\overline{Q}(\mathbf{x}_P, b, ts, oh) = &\sum_{k=1}^{K}\{w_{cc}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{c}_{k,e}\mathbf{c}_{k,f} \\ &+ w_{c2}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N}\mathbf{c}_{k,e}^{2} + w_{pp}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{p}_{k,e}\mathbf{p}_{k,f} \\ &+ w_{p2}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N}\mathbf{p}_{k,e}^{2} + w_{cp}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N}\sum_{f=1,f\neq e}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,f} \\ &+ w_{pc}^{(2)}(b, ts, k, oh)\sum_{e=1}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,e} + w_{c}^{(1)}(b, ts, k, oh)\sum_{e=1}^{N}\mathbf{c}_{k,e} \\ &+ w_{p}^{(1)}(b, ts, k, oh)\sum_{e=1}^{N}\mathbf{p}_{k,e} \\ &+ w^{(0)}(b, ts, k, oh)\}, \;\forall\, b \epsilon A_{OM}, 0 \leq oh \leq OH - 1 \end{aligned} \tag{4.23}$$

where $\overline{Q}(\mathbf{x}_P, b, ts, oh = 0)$ is the estimate of targeted expected value $V_{ts}^{b}(\mathbf{x}_P)$ (see Section 4.2.2.2) for $\forall\,(\mathbf{x}_P, b, ts)$, and $\overline{Q}(\mathbf{x}_P, b, ts, oh)$ is the estimate of targeted expected value $V_{ts}^{a=NAM}(\mathbf{x}_P, b, oh)$ (see Section 4.2.2.2) for $\forall\,(\mathbf{x}_P, b, ts)$ and $1 \leq oh \leq OH - 1$;

(3) if the maintenance choice at time-step $ts$ is either (i) *initiating* a specific overhaul alternative and applying a specific service alternative or (ii) continuing with an undergoing overhaul action and applying a specific service alternative, then

$$\begin{aligned}\overline{Q}(\mathbf{x}_P, a, b, ts, oh) = \sum_{k=1}^{K}\{ & w_{cc}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{c}_{k,e}\mathbf{c}_{k,f} \\ & + w_{c2}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N}\mathbf{c}_{k,e}^2 + w_{pp}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N-1}\sum_{f=e+1}^{N}\mathbf{p}_{k,e}\mathbf{p}_{k,f} \\ & + w_{p2}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N}\mathbf{p}_{k,e}^2 + w_{cp}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N}\sum_{f=1,f\neq e}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,f} \\ & + w_{pc}^{(2)}(a,b,ts,k,oh)\sum_{e=1}^{N}\mathbf{c}_{k,e}\mathbf{p}_{k,e} + w_{c}^{(1)}(a,b,ts,k,oh)\sum_{e=1}^{N}\mathbf{c}_{k,e} \\ & + w_{p}^{(1)}(a,b,ts,k,oh)\sum_{e=1}^{N}\mathbf{p}_{k,e} \\ & + w^{(0)}(a,b,ts,k,oh)\}, \, \forall\, b \epsilon A_{OM}, a \in A_{SM}/\{SM_b\}, 0 \leq oh \leq OH-1 \end{aligned} \tag{4.24}$$

where $\overline{Q}(\mathbf{x}_P, a, b, ts, oh = 0)$ is the estimate of targeted expected value $V_{ts}^{a,b}(\mathbf{x}_P)$ (see Section 4.2.2.2) for $\forall\,(\mathbf{x}_P, a, b, ts)$, and $\overline{Q}(\mathbf{x}_P, a, b, ts, oh)$ is the estimate of targeted expected value $V_{ts}^{a}(\mathbf{x}_P, b, oh)$ (see Section 4.2.2.2) for $\forall\,(\mathbf{x}_P, a \in A_{SM}/\{SM_b\}, b, ts)$ and $1 \leq oh \leq OH - 1$.

For Stage (1):

the quadratic functions of interest for Stage (1) are almost identical to the quadratic functions defined above for Stage (2), and the only different in terms of notations is that *tc* are used to denote the time-step in Stage (1) rather than *ts*; we shall not repeat such functions for Stage (1) here.

**Applying polynomial function method to other case studies**

So far this section discusses why and how to apply the polynomial function method to the power plant case study, including how to reduce the total number of free parameters in the context of the power plant case. Now we generalise the discussion for other case studies.

Regarding other maintenance planning optimisation case studies which follow the same hierarchy (see discussion about the *company-system-asset* hierarchy in Section 4.3) as the power plant case study, in terms of the numerical relationship between the condition/performance of an asset and the expected value in the hybrid MDP model for the corresponding *company*, we expect the existence of concave patterns similar to the ones aforementioned in the power plant case study. Hence we expect the polynomial function method to be a proper modelling choice for such case studies as well. In the power plant case study we detect such concave data patterns by using *average lifetime* and *output amount per time unit* as proxies of the non-observable expected value of interest based on the power plant case study background knowledge; researchers/practitioners may need to adopt/design different proxies accordingly based on their own case study contexts.

The application of the polynomial function to the power plant case study results in the choice of quadratic functions as aforementioned, and such modelling choice is based on numerical test results given specific input data (which we shall specify in Chapter 5) for the case study. Researchers/practitioners may require polynomial functions of higher degrees (compared to quadratic functions) in order to achieve an acceptable value approximation quality in their case studies.

As illustrated above in the context of the power plant case, the application of polynomial functions would result in a large amount of free parameters, and in response we develop a so called *parameters number bounding method* which reduces the amount of free parameters in each quadratic function to a fixed and relatively small number for the power plant case study regardless of the number of the mills per unit. Such method can be applied as part of the polynomial function method to other case studies under one important prerequisite: the *assets* in the corresponding *company* must be identical in terms of their physical structure. In the more general context of *company-system-asset* hierarchy, we expect the application of such *parameters number bounding method* reduces the number of free parameters in each polynomial function to a *constant* which depends on the number of *production systems* and rather than the number of *assets*. As a numerical example, the number of free parameters for each cubic function (suppose we choose cubic functions rather than quadratic functions) in the power plant case study is $23 * K$, regardless of the number of mills per unit. We expect such free parameters reduction effects from the *parameters number bounding method*, based on the assumption that the increment of *assets* amount only results in more polynomial function terms which are homogeneous to the already existing terms (and hence can be defined to share the same free parameters as the already existing terms).

### 4.3.2 Methods interaction and tuning free parameters

In this section we discuss the interaction between the *polynomial function method* and Q-learning in the context of the power plant case (hence we would focus on the quadratic functions defined in Section 4.3.1). We would like to emphasise that the discussion below can be relatively easily generalised for the context of polynomial functions and other aforementioned case studies of interest.

The application of the *polynomial function method* to the power plant case study results in the quadratic functions defined in Section 4.3.1; the update of free parameters in such functions and the value estimates update by Q-learning are intertwined: Q-learning refers to such quadratic functions to obtain the existing value estimates (i.e. the $\overline{Q}$ values) and then Q-learning computes updated value estimates in the way explained in Section 3.3; the quadratic functions refer to such updated value estimates (computed by Q-learning) and accordingly update the free parameters in a *heuristic* way (which we shall specify soon). Such intertwined updates continue for a pre-defined number of iterations. The pre-defined number should be sufficiently large to ensure the output value of the quadratic functions converges.

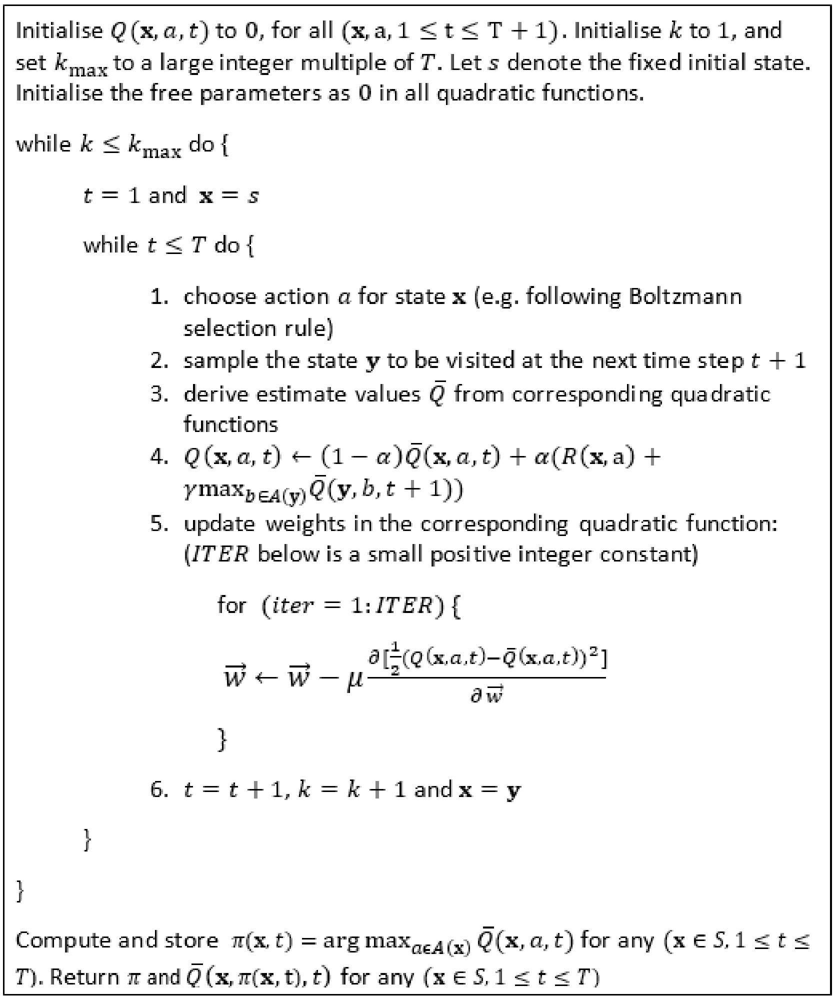
More specifically speaking, the *incremental gradient-descent method* [133] is used to incrementally update the free parameters in such quadratic functions. Again, we shall explain such method in the power plant case study context; we however would like to emphasise such method is in general applicable to tuning free parameters for the data-storage heuristics that follow the *parametric function approximation approach* (see 3.4.2), and relatively good empirical results have been reported for such method in publications (see example publications including [40, 64, 72]).

Let $\overline{Q}$ (the same $\overline{Q}$ values as in Section 4.3.1) denote the existing value estimate of interest; let $Q$ denote the corresponding updated value estimate from Q-learning; let $\vec{w}$ denote the vector of the free parameters in the corresponding quadratic function; the *incremental gradient-descent method* updates the values of the free parameters as follows

$$\vec{w} \leftarrow \vec{w} - \mu \frac{\partial[\frac{1}{2}(Q - \overline{Q})^2]}{\partial \vec{w}}$$

where ← denotes the value calculated on the right-hand side of equation is used to replace the value of the variable on the left-hand side of equation; $\vec{w}$ on the right-hand side represents the values of the free parameters before update whereas $\vec{w}$ on the left-hand side represents the values of the free parameters after update; the value of the parameter $0 < \mu < 1$ is subject to the control of researchers/practitioners who apply the method (see [64] for suggestive controlling rules); $\partial$ denotes partial derivative. In combination with heuristic computation methods such as Q-learning, the *incremental gradient-descent method* aims at approximately minimising the so called *mean squared errors [133]* which is a measure used to evaluate the difference between expected values of interest and their estimates: technically speaking, the *mean squared errors* are the expected sum of the squared value difference between each expected value of interest and the estimate (readers interested in technical details are referred to publications including [64, 118, 133]).

Figure 4.9 provides the pseudo-code regarding how the free parameters tuning and Q-learning (with finite planning-horizon) interacts at a high level.



```
Initialise $Q(\mathbf{x}, a, t)$ to 0, for all $(\mathbf{x}, \mathrm{a}, 1 \leq \mathrm{t} \leq \mathrm{T}+1)$. Initialise $k$ to 1, and
set $k_{\max}$ to a large integer multiple of $T$. Let $s$ denote the fixed initial state.
Initialise the free parameters as 0 in all quadratic functions.

while $k \leq k_{\max}$ do {
        $t = 1$ and $\mathbf{x} = s$
        while $t \leq T$ do {
                1. choose action $a$ for state $\mathbf{x}$ (e.g. following Boltzmann selection rule)
                2. sample the state $\mathbf{y}$ to be visited at the next time step $t+1$
                3. derive estimate values $\bar{Q}$ from corresponding quadratic functions
                4. $Q(\mathbf{x}, a, t) \leftarrow (1-\alpha)\bar{Q}(\mathbf{x}, a, t) + \alpha(R(\mathbf{x}, \mathrm{a}) + \gamma \max_{b \in A(\mathbf{y})} \bar{Q}(\mathbf{y}, b, t+1))$
                5. update weights in the corresponding quadratic function:
                   ($ITER$ below is a small positive integer constant)
                        for $(iter = 1{:}ITER)$ {
                        $\vec{w} \leftarrow \vec{w} - \mu \frac{\partial[\frac{1}{2}(Q(\mathbf{x},a,t)-\bar{Q}(\mathbf{x},a,t))^2]}{\partial \vec{w}}$
                        }
                6. $t = t+1$, $k = k+1$ and $\mathbf{x} = \mathbf{y}$
        }
}
Compute and store $\pi(\mathbf{x}, t) = \arg\max_{a \in A(\mathbf{x})} \bar{Q}(\mathbf{x}, a, t)$ for any $(\mathbf{x} \in S, 1 \leq t \leq T)$. Return $\pi$ and $\bar{Q}(\mathbf{x}, \pi(\mathbf{x}, \mathrm{t}), t)$ for any $(\mathbf{x} \in S, 1 \leq t \leq T)$
```

Figure 4.9: Methods interaction

## 4.4 Conclusion

In this chapter, we apply the performance-centred maintenance approach to the power plant case study. The power plant possesses a multi-level hierarchical physical structure which is beyond the scope of the original mathematical model (see Chapter 2) developed under the performance-centred approach, and therefore we develop a three-stage hybrid MDP model for the case study based on scaling up the existing mathematical model. Additionally we develop a set of heuristics to solve the complex mathematical problem: *two-timescale decomposition method* and *parameters number bounding method,* in combination with the application of *polynomial function method*, *Q-learning* and *incremental gradient-descent method*. The three-stage hybrid MDP model and the set of heuristics can also be used to align the maintenance operations management and production operations management for industrial problems which follow the same hierarchical structure as the power plant case study.

In the next chapter, we shall present and discuss some numerical test results for the power plant case study, in order to numerically demonstrate the practical value of our mathematical model and methods.

# Chapter 5

# Numerical tests

In this chapter, we conduct numerical tests on the power plant case (specified in Section 4.1.4), in order to demonstrate the practical value of our maintenance approach (from Chapter 2), mathematical model and heuristics (both from Chapter 4).

More specifically speaking, we shall first in Section 5.1 compare the performance/effectiveness of heuristics (these are Q-learning, the polynomial function method, the parameters number bounding method and the incremental gradient-descent method) with the performance/effectiveness of exact methods (these are VI and brute-force lookup tables method) in solving a medium size (that is the unit level) problem of the case study. Next, in Section 5.2 we shall further combine such heuristic methods with the two-timescale decomposition method and apply these heuristics to the full size (that is the plant level) problem of the case study, with the purpose of not only deriving near-optimal maintenance decision but also evaluating whether it is more beneficiary for the operators to follow the *low intervention regime* or the *high intervention regime* (specified in Section 4.1.2); additionally, we shall also examine the performance/effectiveness of heuristics in solving the full size problem[1].

[1] The compressed MATLAB Code for the numerical tests at both the unit level and plant level can be found at http://doi.org/10.5281/zenodo.2602872

## 5.1 Unit level: benchmarking heuristic methods

As specified in Section 2.4, we refer to three measures to evaluate the effectiveness of different methods in solving the mathematical problems of interest; a method (or a set of methods) is considered effective in practice only if it satisfies the numerical benchmark specified for each such measure: consuming less than or equal to 24 hours of computational time, requiring less than or equal to 100% of maximum available memory space provided by the standard PC in use, and the maximum data difference between the computational results derived by heuristics and the computational results derived by either exact methods or other benchmarking heuristics is less than or equal to 5%.

In this section, we apply the aforementioned heuristic methods and exact methods to a medium size problem of the power plant case study and examine their performance. The medium size problem consists of a single unit and the corresponding mathematical model contains over $24,000$ unique states.

More specifically speaking, the medium size problem is as follows: the maintainers and operators plan and execute maintenance and production activities respectively as specified in Section 4.1.4 with further simplification assumptions here: the plant contains one unit (which consists of eight mills); the operators increase the work-rate to a fixed high level as long as the number of working mills is less than seven; no penalty cost applies and all produced electricity is sold at a fixed price; both the maintenance planning period and contracted period contain 100 weeks. The parameter values (these are the machine state transition matrices, the maintenance cost of each maintenance alternative, the sales price) of the mathematical model for such medium size problem are obtained from the research of [7] where such values are elicited in a power plant belonging to ScottishPower and furthermore the condition of a mill is categorised into four levels (new, good, poor and failed) and the performance of a mill is categorised into three levels (full performance: that is satisfactory performance; reduced performance: that is unsatisfactory performance;

| Classification | Input parameter | Meaning |
|---|---|---|
| Controlled variables | $T$ | The maintenance planning period |
| | $T_C$ | The contracted period |
| | $K$ | The amount of units in the power plant |
| | $N$ | The amount of mills per unit |
| | $wr$ | The work-rate function |
| | $D_t$ | The demand at time-step $\forall t \leq T$ |
| | $S_M$ | The set of all potential mill states |
| | $A_M$ | The set of all potential maintenance actions at the individual mill level |
| | $OH$ | The amount of time-steps an overhaul action consumes |
| | $SH$ | The amount of time-steps a service action consumes |
| | $C_{SM}$ | The service maintenance cost |
| | $C_{OM}$ | The overhaul maintenance cost |
| | $R_S$ | The sales price per unit of supply |
| | $R_P$ | The penalty cost per unit of unfulfilled contracted supply |
| | $P_M$ | Mill state transition matrix |
| | $\gamma$ | One time-step discount ratio |
| Independent variables | $X_U$ for $t = 1$ | The initial unit state |

Table 5.1: Parameter classification in medium size problem

offline: that is grossly unsatisfactory performance); such parameter values are business sensitive information and therefore we shall not present them in this thesis. Such single unit problem is part of the research in [8] and they solve this problem by applying exact methods (VI and brute-force lookup tables method) on a high performance computer. Note here we instead implement such exact heuristics on a standard PC (specified below) for the purpose of performance comparison.

The numerical tests of this PhD project are conducted in MATLAB R2014a on a PC with the following properties: processor: Intel(R) Core(TM) i5-3570 CPU@3.40GHz; operating system: Windows 7 Enterprise 32-bit Operating System; RAM: 8 GB.

### 5.1.1 Performance of heuristics and exact methods

We classify the input parameters of the mathematical model into two types: controlled variables (the values/settings of which stay constant in the numerical experiment here) and independent variables (the values/settings of which we manipulate), as shown in Table 5.1.

In order to thoroughly benchmark the heuristics, we select ten different initial states (presented in Table 5.2 where each unit state is expressed as a set consisting of the state index of all eight mills in the unit, and readers are referred to Table 4.1 in Section 4.1.4 for the meaning of mill state index) for the mathematical model and ensure such ten initial states in the mathematical model represent different potential overall physical status of the unit in reality ranging from relatively good physical status to relatively bad physical status; we consequently generate ten different MDP problems each of which has a unique initial state, and for each such MDP problem we benchmark the performance of the heuristics. In order to facilitate the discussion, we index such ten MDP problems from 1 to 10 as shown in Table 5.2.

| **MDP problem index** | **Initial unit state** (in terms of mill state index) |
|---|---|
| 1 | $\{10, 10, 10, 10, 10, 10, 10, 10\}$ |
| 2 | $\{10, 10, 10, 10, 10, 10, 10, 9\}$ |
| 3 | $\{10, 10, 10, 10, 9, 9, 9, 8\}$ |
| 4 | $\{10, 9, 9, 9, 9, 9, 8, 7\}$ |
| 5 | $\{10, 10, 10, 10, 8, 8, 7, 6\}$ |
| 6 | $\{10, 10, 9, 9, 8, 7, 6, 5\}$ |
| 7 | $\{10, 10, 10, 10, 10, 10, 9, 4\}$ |
| 8 | $\{10, 10, 10, 10, 9, 9, 9, 3\}$ |
| 9 | $\{10, 10, 10, 10, 10, 9, 8, 2\}$ |
| 10 | $\{10, 10, 10, 10, 10, 10, 8, 1\}$ |

Table 5.2: Initial unit state

In the context of the single unit problem discussed above, the application of our heuristics should ideally contain a large number of iterations (see specification in Section 4.9) to ensure the optimal value estimate converges for *each* state-time-step in the MDP problem, in order to obtain a near-optimal policy which prescribes a near-optimal action to each state-time-step in the MDP problem. In practice, however, we realise such convergence goal imposes an impractically large number of iterations and hence an impractically long computational time which defeats the purpose of introducing heuristics; we therefore alternatively choose to only require the convergence of the optimal value estimation for the *initial* state in each MDP problem. As a result, we expect the heuristics can relatively quickly derive a highly accurate optimal value estimate and a near optimal maintenance action for the *initial* state. In other words, for each such ten MDP problem, the application of the heuristics aims at deriving a near-optimal maintenance action for the *initial* state within a relatively short computational time rather than spending an impractically long computational time deriving a near-optimal maintenance policy for an entire MDP problem of the case study. Note such strategy of applying heuristics still serves the purpose of solving real life decision making problems: the practitioners simply need to update the initial state of the mathematical model as the unit physical status evolves in practice (and consequently obtain a new MDP problem) and then re-apply the heuristics to the new MDP problem to derive updated results.

Below we present the numerical test results and compare the performance between heuristics and exact methods against the three aforementioned numerical benchmarks; furthermore, based on some additional empirical results we shall also discuss why the heuristics are able to derive near optimal results for the initial state of an MDP problem *without* requiring the convergence of optimal value estimation for all the other state-time-steps in the MDP problem.

**Computational time: convergence speed**

For each MDP problem, the application of heuristics contains $100,000$ iterations which consume approximately 50 minutes in total. The step-size parameter $\alpha$ in Q-learning is controlled by the rule $\alpha = \frac{M}{N+k}$ as discussed in Section 3.3, and we set $M = 11,250,000$ and $N = 12,500,000$; the other step-size parameter $\mu$ in the incremental gradient-descent method is set as $0.2/iter$, and $ITER = 2$ (see specifications in Figure 4.9 of Section 4.3.2). Such parameters setting in the application of heuristics is based on our domain knowledge of the project and trial-and-error in a toy-size problem (which contains only three mills). The action selection in such application of heuristics (more specifically speaking, Q-learning) is controlled by pure exploration policy (specified in Section 3.3.1).

For each MDP problem, the optimal value estimation converges in less than 20 minutes. For instance, Figure 5.1 illustrates how the optimal value estimation converges for the initial state of MDP problem 10 in simulation, in terms of $\frac{\text{optimal value estimate}}{\text{optimal value}}$.

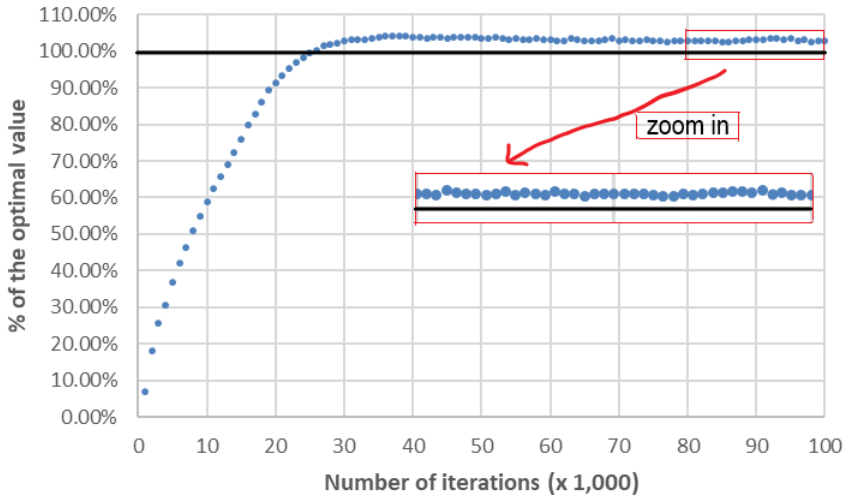


Figure 5.1: Optimal value estimation (% of the optimal value) in every 1,000 iterations for initial state of MDP problem 10

As comparison, for each MDP problem, the exact methods would consume more than 10 days! Such conclusion is drawn as follows: first we observe the computational time spent on performing a fraction of the total computational workload; then we use such observed fractional computational time to extrapolate the total computational time, and such total computational time is estimated to be more than 10 days; finally we examine such estimation by keeping the exact methods running on the standard PC for exactly 10 days, and the numerical test confirms it is indeed impossible to finish the total computational workload in 10 days.

**Data-storage cost**

Numerical tests show that both heuristics and exact methods are effective in terms of their data-storage costs. However, in terms of storing the computational results, the brute-force methods consume approximately 70% of the maximum available memory space provided by the standard PC in use: we draw such conclusion by incrementally enlarging the problem size (more specifically speaking, the time length of planning period) until the data-storage cost of brute-force methods surpasses the maximum available memory space of the PC and then comparing the original problem size with the enlarged problem size; the data-storage cost of heuristics is approximately only 0.04% of the data-storage cost induced by brute-force methods; additionally, the data-storage cost of brute-force methods grows exponentially with the total number of mills in the industrial problem, while the data-storage cost of heuristics does not depend on the total number of mills (as discussed in Section 4.3.1) and hence is immune from such exponential growth.

**Data accuracy level**

Table 5.3 summarises the data accuracy level of heuristics for such ten MDP problems by comparing with the exact computational results (note the exact results are derived by [8] via implementing exact methods on a high performance computer): technically speaking, Column $\frac{\text{Optimal value estimate}}{\text{Optimal value}}$ describes what percentage the optimal value estimate (derived from heuristics) is of the exact optimal value (derived from exact methods) for the initial state of each MDP problem (intuitively speaking, such column describes how accurate the optimal value estimate is; the closer the percentage value is to 100% the more accurate the estimate is, and a percentage value larger or smaller than 100% in this column indicates overestimate or underestimate respectively); Column *Selected action index (optimal or not)* describes the estimated optimal action (derived from heuristics) for the initial state of each MDP problem, and whether the estimation is accurate or not (note that we index different maintenance alternatives in the mathematical model (from Chapter 4) and here in the column we refer to each action by its index number); additionally, Column $\frac{\text{Expected value}}{\text{Optimal value}}$ describes what percentage the expected value of the action selected from heuristics is of the optimal value (both such values are derived from exact methods), for the initial state of each MDP problem (intuitively speaking, such column measures how good the action derived by heuristics is compared with the optimal action; the closer the percentage value is to 100% the better).

According to Table 5.3, the estimation error of the optimal value falls within the ±5% range for the initial state of each MDP problem (see Column $\frac{\text{Optimal value estimate}}{\text{Optimal value}}$);

the optimal action estimation for the initial state is accurate for the majority (nine out of ten) of MDP problems; regarding the only MDP problem where the optimal action estimation is not accurate for the initial state (i.e. MDP problem 4), the loss of monetary value due to the selection of a sub-optimal action (i.e. action 15) is merely 1.6% compared to selecting the actual optimal action. In conclusion, the approximation results derived from the heuristics are relatively accurate (see discussion on benchmarks in Section 2.4) for the single unit problem.

| MDP problem index | $\frac{\text{Optimal value estimate}}{\text{Optimal value}}$ | Selected action index (optimal or not) | $\frac{\text{Expected value}}{\text{Optimal value}}$ |
|---|---|---|---|
| 1 | 98.68% | 1 (optimal) | 100.00% |
| 2 | 99.49% | 16 (optimal) | 100.00% |
| 3 | 99.22% | 15 (optimal) | 100.00% |
| 4 | 95.31% | 15 (not optimal) | 98.4% |
| 5 | 96.98% | 13 (optimal) | 100.00% |
| 6 | 95.79% | 12 (optimal) | 100.00% |
| 7 | 98.86% | 11 (optimal) | 100.00% |
| 8 | 98.51% | 10 (optimal) | 100.00% |
| 9 | 99.03% | 9 (optimal) | 100.00% |
| 10 | 102.83% | 8 (optimal) | 100.00% |

Table 5.3: Data accuracy level of heuristics

Further examination of the numerical results in Table 5.3 indicates an interesting pattern: as shown in Column $\frac{\text{Optimal value estimate}}{\text{Optimal value}}$, the heuristics tend to underestimate the optimal values. We believe such issue occurs as follows: in this case study, there exits no ideal concave curve defined by the quadratic functions (specified in Section 4.3.1) around which the scatter of expected values (defined in Section 3.3) strictly follows a normal distribution: in other words, some outliers exist; in such context, using the incremental gradient-descent method (discussed in Section 4.3.2) to tune the free parameters of quadratic functions would encounter an issue: such method aims at approximately minimising the mean squares of the value distances between the expected values and their estimates (produced by such quadratic functions), but the outliers would affect and even dominate

| **Performance measure** | **Are heuristics effective?** | **Are exact methods effective?** |
|---|---|---|
| Computational time | Yes | No |
| Data-storage cost | Yes | Yes |
| Data accuracy level | Yes | Yes |

Table 5.4: Medium size problem: performance comparison between heuristics and exact methods

the calculation of such mean squares and therefore induce either overestimation or underestimation of most values (in this case study it is underestimation). In order to mitigate such underestimation tendency, one future research direction would be either improving the incremental gradient-descent method or designing a new parameters tuning method to ensure a certain level of "robustness" to the outliers when tuning up free parameters.

Before discussing why the heuristics are able to derive near optimal results for the initial state of an MDP problem *without* requiring the convergence of optimal value estimation for all the other state-time-steps in the MDP problem, here in Table 5.4 we summarise the comparison between heuristics and exact methods on each of the three aforementioned performance measures: in the medium size problem of case study, the numerical test results indicate that the heuristics are effective in solving the mathematical problems of interest, whereas the exact methods are not effective in terms of the computational time.

**Convergence for initial state and the implications**

First of all, we would like to emphasise that in the numerical tests discussed above the optimal value estimation does *not* converge for all state-time-steps in the ten MDP problems. For instance, for MDP problem 1, after the limited number of iterations (i.e. $100,000$ iterations) in simulation, the heuristics visit (see terminology in Section 3.3.1) $1,918$ different states in total at the $15^{\text{th}}$ week, and the optimal value estimation converges for 36.39% (rather than 100%) of such $1,918$ states at such time-step; additionally, the heuristics visit $6,443$ different states in total at the last time-step (i.e. the $100^{\text{th}}$ week), and the optimal value estimation converges for 24.02% (rather than 100%) of such $6,443$ states at the last time-step. Similar examination (that is checking the number of states visited by the heuristics and checking the percentage of states with converged optimal value estimation) at each time-step for all aforementioned ten MDP problems indicates the following data patterns: the further away the time-step is (towards the planning-horizon), the larger the total number of states are visited by the heuristics at such time-step while the optimal value estimation converges for a smaller percentage of such states: the aforementioned comparison between the 15th week and the 100thweek of MDP problem 1 is an illustrative example of such data patterns.

Below we shall discuss why the heuristics are able to derive near optimal results for the initial state of an MDP problem *without* requiring the convergence of optimal value estimation for all the other state-time-steps in the MDP problem, and we shall also discuss why the aforementioned data patterns exist in the numerical tests.

Increment of states visited per time-step

A crucial part of the heuristics (more specifically speaking, Q-learning) is simulating the system state evolving process for MDP problems in general (as specified in Chapter 3), and additionally in the mathematical model the number of states that can potentially branch out at each time-step following the initial unit state increases as the time-step continues towards the planning-horizon; it is therefore reasonable that in the numerical tests the total number of states visited by heuristics increases at each time-step as the time-step continues towards the planning-horizon in each MDP problem.

Convergence of state optimal value(s) estimation per time-step

Intuitively speaking, the optimal value estimate of each state branched out at later time-steps in the mathematical model carries a certain weight in terms of impacting the optimal results estimation accuracy of the initial state in the application of heuristics; however, given the distributions of the random variables which govern the system dynamics (see specifications in Section 3.3.1), the aforementioned weights vary among such branched out states: more specifically speaking, the less likely a state can potentially branch out from the initial state at a given time-step, the smaller such weight the state carries compared to other more likely branched out states at the same time-step. Therefore at each time-step in the MDP problem only a limited portion (rather than 100%) of branched out states are relatively important in terms of impacting the optimal results estimation accuracy level of the initial state in the application of heuristics.

Furthermore, the heuristics (more specifically speaking, Q-learning) prioritise driving the optimal value estimates of such relatively important states at each time-step to converge towards the optimal values by visiting (see terminology in Section 3.3.1) such states more frequently (such frequency is underpinned by the distributions of the random variables which govern the system dynamics; see specifications about the state visiting/sampling in Section 3.3.1). Intuitively speaking, the more frequently a state is visited at a time-step during simulation (i.e. in the application of heuristics), the more likely the optimal value estimation would converge for such state-time-step compared to other states at such time-step of the MDP problem (readers who are interested in technical discussion are referred to publications including [64, 72] which justify such intuition by relating to the so called *Robbins-Monroe stochastic approximation scheme*).

Given such properties discussed above regarding the MDP problem and the heuristics (more specifically speaking, Q-learning), it is reasonable that (1) the optimal value estimates only converge for a limited portion (rather than 100%) of branched out states at each time-step in the simulation given a limited number of iterations and (2) the heuristics are able to derive near optimal results for the initial state of an MDP problem *without* requiring the convergence of optimal value estimation for all the other state-time-steps in the MDP problem.

Additionally, the existence of discount factor in the mathematical model renders such aforementioned impact of later branched out states in general less crucial compared to states branched out at earlier time-steps in the mathematical model; therefore it is reasonable to assume in the MDP problem the percentage of such aforementioned relatively important states in general decreases at each time-step as the the time-step continues towards the planning-horizon, and hence it is reasonable that in the numerical tests the percentage of states with converged optimal value estimation decreases at each time-step as the time-step continues towards the planning-horizon.

Further discussion

Here we would like to discuss two additional questions on the convergence performance of the heuristics in terms of speed and accuracy.

*How would the convergence behaviour (discussed above) impacted in general if the simulation contains more iterations (in other words, what if the application of heuristics runs longer)?* Intuitively speaking, as the total number of iterations increases in the simulation (for example $500,000$ iterations rather than $100,000$ iterations), the percentage of states with converged optimal value estimation would become higher at each time-step (for example such percentage might be 60% rather than only 36.39% as reported above at the 15th week of MDP problem 1); in an extreme case, if the total number of iterations increases to a very large value that is close to infinity (and of course the computational time would be impractically long), such percentage would become close to 100% at each time-step and as a result the application of heuristics would derive a near optimal maintenance policy (for the entire MDP problem) rather than only derive a near optimal maintenance action for the initial state of the MDP problem.

As discussed above, we recommend the following heuristics application approach to practitioners: update the initial state of the mathematical model as the unit physical status evolves in reality and re-apply the heuristics; in the application of heuristics, only require convergence of optimal value estimation for the initial state of the mathematical problem. By following such approach, the necessary computational time is expected to be relatively short in order to obtain converged results of interest, and such expectation is indeed supported by the empirical results discussed above. *However, is it reasonable to* *expect that the heuristic results would still converge relatively quickly if the time length of the planning period in the mathematical model becomes infinitely large?*

Here we would like to remind readers a main difference between applying the heuristics (more specifically speaking, Q-learning) to a finite-horizon problem and to an infinite-horizon problem: in a finite-horizon problem setting, the heuristics *return* to the initial state of interest *once* the sampling process reaches the last time-step of the mathematical model and then the heuristics would again sample the states that can potentially branch out from the initial state at each time-step of the mathematical model; in an infinite-horizon problem setting, the simulation is *not* defined to return to the initial state of interest after sampling the state evolving process for a fixed number of time-steps *any more* (readers can refer back to the methodology of *Q-learning for infinite-horizon MDPs* in Section 3.3.2 in case of requiring further clarification). Actually, given the relatively large size of the state space, in an infinite planning-horizon problem setting, the simulation may take a relatively long computational time to return to the initial state of interest and then again sample the states that can potentially branch out from the initial state. In the worst potential scenario, the simulation may roam through the entire state space multiple times before it returns to the initial state. Such return-sample process however must be repeated for a relatively large number of times to derive the heuristic results to converge. In other words, in an infinite planning-horizon problem setting, given a relatively large state-space, the total computational time may become impractically long in order to derive converged results, even if the heuristics discussed above are used (and of course in comparison the computational time of applying exact methods would be even longer).

We note that in some other studies (such as [56, 65, 66, 67, 68, 70, 71, 72, 119, 131, 151]) Q-learning and other reinforcement learning methods (especially if they are further scaled up with value function approximation heuristics) are reported to derive near-optimal results within a relatively short computational time for relatively large size

MDP problems with *infinite* planning-horizon (see discussion in Section 3.3); we believe it is because in such studies under the optimal policy the MDP model would only evolve into a relatively small number of states (rather than all states or majority of all states) with high probabilities, and hence most of the time the simulation is more likely to roam within a relatively small size sub-set of states rather than roam through the entire state-space or majority of the entire state-space (readers can refer back to Chapter 3 in case of requiring further explanation). Such nice property regarding the state space however does not seem to hold in our case study, as the computational time becomes impractically long to derive converged heuristic results for the aforementioned ten MDP problems if we re-model the planning periods of such problems as infinitely long. Additional heuristics (such as the two-timescale decomposition method developed in Chapter 4) are therefore required to ensure the computational time is relatively short to derive converged results for infinite-horizon problems.

Two-timescale decomposition method

In Section 5.2 we shall conduct numerical tests on the full size problem of the case study where we assume an *infinite* planning-horizon problem setting (as discussed in Chapter 4). In order to ensure the heuristic results converge in a relatively short time for the full size problem, we shall further combine the heuristics already examined above in the medium size problem with the two-timescale decomposition method (see Chapter 4).

Note that the two-timescale decomposition method would inevitably compromise the accuracy of the heuristic results, and in the future research we shall revisit the medium size problem and further examine the accuracy level of the heuristic results when the two-timescale decomposition method is used.

## 5.2 Plant level: comparison between different operators' decision regimes

In this section, we apply our maintenance approach and heuristics to the full size problem of the case study, with the purpose of illustrating that our maintenance approach and mathematical model facilitate both optimal maintenance decision making and machine utilisation behaviour improvement in complex problem settings which follow the general hierarchical structure specified in Section 4.1.3; additionally, we also examine the performance/effectiveness of the heuristics in solving the full size problem.

First we shall in Section 5.2.1 explain the numerical test setting; then we shall in Section 5.2.2 present and discuss the numerical test results.

### 5.2.1 Numerical test setting

Here we consider the full size (that is the plant level) problem of the case study. The power plant contains four units each of which consists of eight mills. In the full size problem, we reuse the parameter values (these are the machine state transition matrices, the maintenance cost of overhaul and service, the sales price per unit of supply) from the medium size problem discussed in Section 5.1. Additionally, we assume the following: the contracted period contains 20 weeks, the short-term part of the non-contracted period contains 80 weeks and the long-term part of the non-contracted period is infinitely long. Furthermore, we assume the demand of each unit in each week within the non-contracted period follows the same continuous uniform distribution unif$(a, b)$ where $a$ equates to 70% of the maximum weekly production rate of a perfect unit working at normal work-rate and $b$ equates to 100% of such maximum weekly production rate; regarding the contracted period, we specify the contracted demand of each unit in each week by randomly sampling from the aforementioned uniform distribution, and additionally we assume the penalty

| Classification | Input parameter | Meaning |
|---|---|---|
| Controlled variables | $T$ | The maintenance planning period |
| | $T_L$ | The long-term part of the non-contracted period |
| | $T_s$ | The short-term part of the non-contracted period |
| | $T_C$ | The contracted period |
| | $K$ | The amount of units in the power plant |
| | $N$ | The amount of mills per unit |
| | $H_{SM}$ | The set of the time-steps where the maintenance alternatives include service in Stage (3) of the model; specified based on the deterministic rotating schedules and permutation rule in Section 4.2.1.1. |
| | $\boldsymbol{D}_t$ | The vector that consists of the demand (distribution) at time-step $\forall t \leq T$ for unit $(1, 2, ..., K)$ |
| | $S_M$ | The set of all potential mill states |
| | $A_M$ | The set of all potential maintenance actions at the individual mill level |
| | $OH$ | The amount of time-steps an overhaul action consumes |
| | $SH$ | The amount of time-steps a service action consumes |
| | $C_{SM}$ | The service maintenance cost |
| | $C_{OM}$ | The overhaul maintenance cost |
| | $R_S$ | The sales price per unit of supply |
| | $R_P$ | The penalty cost per unit of unfulfilled contracted supply in Stage (1) of the model |
| | $P_M$ | Mill state transition matrix |
| | $\gamma$ | One time-step discount ratio |
| Independent variables | $wr$ | The work-rate function that applies to any unit at any time-step in Stage (1) of the model |
| | $X_P$ for $tc = 1$ | The initial plant state |

Table 5.5: Parameter classification in full size problem

cost per unit of demand is equal to 30% of the sales price.

We classify the input parameters of the mathematical model into two types: controlled variables (the values/settings of which stay constant in the numerical experiment here) and independent variables (the values/settings of which we manipulate), as shown in Table 5.5.

For the purpose of illustrating the practical value of our maintenance approach and heuristics, we randomly select some different initial power plant states for the three-stage hybrid MDP model (defined in Section 4.2.2) and further embed the mathematical model with certain intervention regime (that is *low/high intervention regime* discussed in Section 4.1.2); we consequently generate different three-stage hybrid MDP problems each of which has a unique combination of initial plant state and intervention regime. Furthermore, we ensure such problems are generated in pairs: each problem associated with a certain initial state and *low/high intervention regime* is paired with a different problem associated with the same initial state but *high/low intervention regime* (that is the opposite intervention regime) as counterpart.

We then apply the aforementioned set of heuristics to each such problem in order to derive optimal value estimation and optimal maintenance choice estimation for the initial plant state of the problem given certain intervention regime is embedded; furthermore, by comparing such optimal value estimate for the same initial plant state between different regimes, operators can decide which regime brings better costs/benefits given specific initial plant state.

In total ten hybrid MDP problems are examined: we randomly choose five different initial plant states (presented in Table 5.6 where a plant state is expressed as a set comprised of the states of all four units each of which is further expressed as a set consisting of the state index of all eight mills in the unit, and readers are referred to Table 4.1 in Section 4.1.4 for the meaning of mill state index; note the five initial plant states in the mathematical model represent different potential overall physical status of the plant in reality ranging from relatively good physical status to relatively bad physical status) and combine each such state with the two intervention regimes of interest; then we assign the combinations to the three-stage hybrid MDP model and sequentially generate ten different problems which are grouped into five pairs. In order to facilitate later discussion, we index such ten problems in a format as illustrated in Table 5.7: for instance, problem L3 and problem H3 are a pair of problems; problem L3 is embedded with the *low intervention regime* and initial plant state 3, and the counterpart problem is problem H3.

### 5.2.2 Numerical results and discussion

The state space of the full size problem consists of more than $10^{16}$unique states at the plant level of the power plant case study, which is over $4 * 10^{11}$times of the state space size of the medium size problem discussed in Section 5.1; additionally, the action space

| **Initial plant state index** | **{unit 1 state, unit 2 state, unit 3 state, unit 4 state}** |
|---|---|
| 1 | $\{\{10, 10, 10, 10, 10, 10, 10, 9\},$<br>$\{10, 10, 10, 10, 9, 9, 9, 8\},$<br>$\{10, 9, 9, 9, 9, 9, 8, 7\},$<br>$\{10, 10, 9, 9, 8, 7, 6, 5\}\}$ |
| 2 | $\{\{10, 10, 10, 10, 9, 9, 9, 3\},$<br>$\{10, 10, 10, 10, 8, 8, 7, 6\},$<br>$\{10, 10, 10, 10, 10, 10, 10, 10\},$<br>$\{10, 10, 10, 10, 9, 9, 9, 8\}\}$ |
| 3 | $\{\{10, 10, 10, 10, 10, 10, 8, 1\},$<br>$\{10, 10, 10, 10, 10, 10, 10, 9\},$<br>$\{10, 10, 10, 10, 10, 9, 8, 2\},$<br>$\{10, 10, 9, 9, 8, 7, 6, 5\}\}$ |
| 4 | $\{\{10, 10, 10, 10, 10, 10, 8, 1\},$<br>$\{10, 10, 10, 10, 10, 10, 10, 10\},$<br>$\{10, 10, 10, 10, 8, 8, 7, 6\},$<br>$\{10, 10, 9, 9, 8, 7, 6, 5\}\}$ |
| 5 | $\{\{10, 10, 10, 10, 10, 10, 10, 10\},$<br>$\{10, 10, 10, 10, 9, 9, 9, 3\},$<br>$\{10, 10, 10, 10, 10, 9, 8, 2\},$<br>$\{10, 10, 10, 10, 10, 9, 8, 2\}\}$ |

Table 5.6: Initial plant state

size of the full size problem is over 10 times of the action space size of the medium size problem.

It is infeasible to implement exact methods to solve the full size problem, as exact methods induce an impractically long computational time and an impractically large data-storage cost: for instance, the aforementioned set of brute-force methods (these are VI and lookup tables method) is estimated to take more than $4 * 10^{13}$ days to solve the full size problem, based on the performance of such methods in the medium size problem (note such estimation is derived based on a simplification assumption that it takes the same amount of time to calculate the expected value of a state-action-time-step in the full size problem as the medium size problem; however in practice such computational time is actually higher in the full size problem, as each state-action pair can potentially evolve into more states at the following time-step in the hybrid MDP model for the full size problem compared to the medium size problem. In other words, the actual total computational time induced by brute-force methods in solving the full size problem may be even several magnitudes larger than $4 * 10^{13}$ days); additionally, in terms of storing the computational

| Problem index | Initial plant state index | Intervention regime (low/high) |
|---|---|---|
| L1 | 1 | low |
| H1 | 1 | high |
| L2 | 2 | low |
| H2 | 2 | high |
| L3 | 3 | low |
| H3 | 3 | high |
| L4 | 4 | low |
| H4 | 4 | high |
| L5 | 5 | low |
| H5 | 5 | high |

Table 5.7: Different three-stage hybrid MDP problems

results, the application of such brute-force methods approximately induces $4 * 10^{12}$ times of data-storage cost in solving the full size problem compared to the medium size problem, and such large data-storage cost is approximately $3 * 10^{12}$ times of the maximum available memory space of the standard PC in use. In summary, it is necessary to apply heuristics to solve the full size problem.

For each of the ten hybrid MDP problems, the application of heuristics contains $300,000$ iterations which consume approximately 9.5 hours in total, and the optimal value estimation converges in approximately 7.5 hours. For instance, Figure 5.2 illustrates how the optimal value estimation converges for the initial state of problem H2 in simulation, in terms of $\frac{\text{optimal value estimate}}{\text{final estimate}}$.

The two-timescale decomposition method plays a crucial role in ensuring the heuristic results converge in such relatively short computational time (approximately 7.5 hours): due to the application of such heuristic, the long-term part of the decision making problem is solved by brute-force methods in a short time (see specification in Chapter 4); the associated computational results are further fed into the simulation (via Equation (4.20) defined in Chapter 4) and therefore help the simulation "leap" to a solution that is relatively close to optimum: as highlighted in Figure 5.2, the convergence rate of simulation for problem H2 is obviously improved in the early stage of simulation, and similar improvement on convergence speed is also observed in the simulation for all the other representative hybrid MDP problems; in contrast, such convergence speed improvement disappears and the required computational time becomes impractically long (to derive converged results) if the two-timescale decomposition method is not implemented, as we observe in some additional comparative numerical tests.

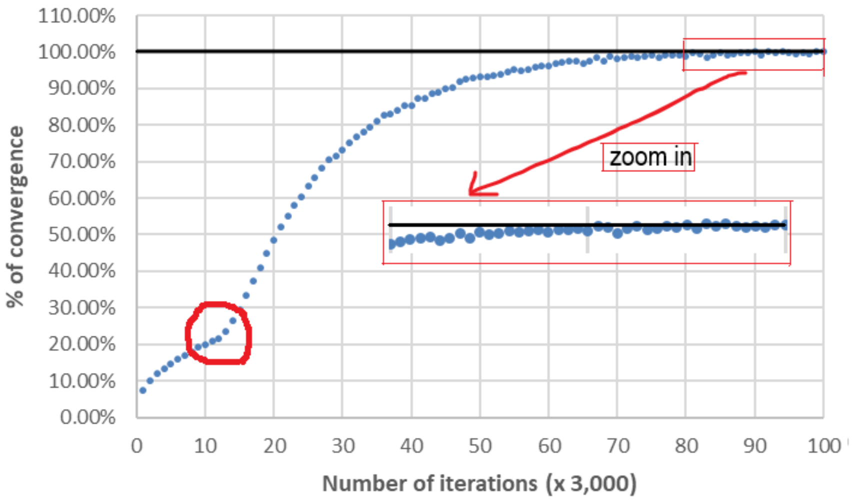


Figure 5.2: Optimal value estimate (% of the final estimate) in every 3,000 iterations for initial state of problem H2

In terms of storing the computational results, the data-storage cost induced by heuristics in the full size problem is only 4 times of the data-storage cost in the medium size problem, and it approximately only consumes 0.11% of the maximum available memory space of the standard PC in use.

In terms of benchmarking the data-accuracy level, we do not have exact results or heuristic results derived by other benchmarking heuristics to compare with. However, we assume the heuristics (these are Q-learning, the polynomial function method, the parameters number bounding method and the incremental gradient-descent method) examined in the medium size problem would reach approximately the same level of accuracy here in the full size problem, given that the medium size problem resembles the full size problem in terms of the dynamics nature (i.e. the stochastic machine deterioration process and

| **Performance measure** | **Are heuristics effective?** | **Are exact methods effective?** |
|---|---|---|
| Computational time | Yes | No |
| Data-storage cost | Yes | No |

Table 5.8: Full size problem: performance comparison between heuristics and exact methods

work-rate adjustment flexibility); additionally, in terms of the two-timescale decomposition method which is used in solving the full size problem, its impacts on the data accuracy level should be examined in future studies as discussed in Section 5.1, but we assume such impacts are marginal: such method applies to the long-term part of the non-contracted period of the mathematical model, and the importance of such long-term part on calculating the expected values is limited due to the discount factor in the mathematical model.

Before presenting and comparing the numerical results between different intervention regimes, in Table 5.8 we summarise the performance/effectiveness of heuristics and exact methods in solving the full size problem of the case study.

Table 5.9 summarises the numerical results for different intervention regimes: Column *Maintenance choice index* $(a, b)$ describes the estimated optimal maintenance choice for the initial plant state of each problem (note that we index different maintenance alternatives in the three-stage hybrid MDP model by specifying two integer valued variables $(a, b)$, as explained in Section 4.2.2.2, and here in the column we refer to each maintenance choice by its index); Column $\frac{\bar{Q}_L}{\bar{Q}_H}$ compares the estimated optimal values for each initial plant state between every two paired problems (technically speaking such column describes what percentage the optimal value estimate for the initial state in a problem embedded with low intervention regime is of the optimal value estimate for the same initial state in the counterpart problem embedded with high intervention regime, and intuitively speaking such column indicates which regime brings better profits given certain initial plant state where a percentage value larger or smaller than 100% indicates the low intervention regime or high intervention regime should be selected respectively).

| Problem index | Maintenance choice index $(a, b)$ | $\frac{\bar{Q}_L}{\bar{Q}_H}$ |
|---|---|---|
| L1 | $(12, 17)$ | 99.79% |
| H1 | $(12, 21)$ | |
| L2 | $(3, 23)$ | 99.95% |
| H2 | $(3, 11)$ | |
| L3 | $(1, 23)$ | 100.2% |
| H3 | $(1, 23)$ | |
| L4 | $(1, 5)$ | 101.28% |
| H4 | $(1, 5)$ | |
| L5 | $(29, 13)$ | 99.77% |
| H5 | $(20, 23)$ | |

Table 5.9: Comparison between different intervention regimes

As indicated by Table 5.9, the two intervention regimes are expected to yield almost identical profits for each given initial plant state in practice: although Column $\frac{\bar{Q}_L}{\bar{Q}_H}$ suggests the high intervention regime brings slightly higher profits given initial plant state 1, 2 or 5 whereas the low intervention regime brings slightly higher profits given initial plant state 3 or 4, the value difference (in terms of percentage) between the two regimes revealed by Column $\frac{\bar{Q}_L}{\bar{Q}_H}$ is too small to support such claim conclusively given that heuristic results are used in comparison rather than exact results. In other words, both regimes are expected to be approximately equally beneficiary for operators to follow given any initial plant state selected above.

Such what-if numerical tests (on different operators' decision regimes) are conducted following the problem setting specified above for the power plant case study; researchers/practitioners can apply our maintenance approach, mathematical model and heuristics to the problem setting of their own cases and conduct similar what-if analysis to facilitate operations decision making for both maintainers and operators.

## 5.3 Conclusion

In this chapter we examine the practical value of our maintenance approach, mathematical model and heuristics in the power plant case study. As demonstrated in the numerical tests, our maintenance approach and mathematical model support decision making of both maintainers and operators from a balanced view between machine utilisation and increased risk of failure; additionally, such facilitation on optimising maintenance planning and improving machine utilisation behaviours also applies to complex industrial problems which follow the general hierarchical structure specified in Section 4.1.3. Furthermore, the numerical test results of the case study empirically illustrate the high effectiveness of our heuristics by benchmarking with exact methods on three measures: computational time, data-storage cost and data accuracy level; such numerical test results indicate that the performance of our heuristics are robust to the size of mathematical problems of interest.

In the next chapter, we shall summarise our research contributions and discuss future research directions.

# Chapter 6

# Conclusions and further research

This thesis focuses on facilitating the balance between machine utilisation and increased risk of failure in a generic manufacturing business setting extracted from a large scale British coal-fired power plant; the key features of such business setting include (1) machine maintenance and machine utilisation are tightly intertwined and (2) maintainers and operators follow different decision making time-scales.

Chapter 1 highlights the research questions investigated in this thesis:

- How to balance between machine utilisation and increased risk of failure in the aforementioned business setting?
- How to scale up the mathematical model from Chapter 2 to facilitate maintenance planning optimisation in a more complex business setting which further involves the hierarchical structure defined in Chapter 4?
- How to effectively solve the large size mathematical problems resulted from applying our maintenance approach (developed in Chapter 2), knowing such mathematical problems would induce (1) impractically long computational time and (2) impractically large data-storage cost if brute-force methods are applied?

Here we summarise our research contributions which are made in response to the above research questions in sequence:

Building a new performance-centred maintenance approach

As discussed in Chapter 2, the existing maintenance approaches cannot effectively resolve the maintenance planning optimisation problem in the above business setting from a balanced view between machine utilisation and increased risk of failure, and such issue is common in the manufacturing industry; in order to fill in such an important gap between the existing literature and the research problem of interest, a new maintenance approach is developed in Chapter 2 to properly capture the impacts of different decision making time-scales and also to balance between machine utilisation and increased risk of failure.

Such research contribution, as specified in Chapter 2, is achieved through (1) investigating why maintainers and operators have different decision making time-scales and how such time-scale distinction impacts the value-perception difference between maintainers and operators regarding various operations activities, and then based on the investigation results (2) conceptually eliciting how the decision making of maintainers and operators are intertwined, and finally based on the elicitation results and a set of modelling choices (3) developing a mathematical model to capture the maintenance planing problem of interest.

Furthermore, as discussed in Chapter 2, by integrating the operators' decision regime into maintainers' decision making in the modelling work, the resulted maintenance approach facilitates not only maintenance planning optimisation but also machine utilisation behaviour improvement from a balanced view between machine utilisation and increased risk of failure. In other words, such a maintenance approach facilitates decision making for both managerial parties and align their business functions more effectively for better costs/profits in the aforementioned business setting.

Developing a new mathematical model to capture the hierarchical structure of interest

As discussed in Chapter 4, further investigation of the specific power plant case background highlights another issue: industries that have adopted distributed generation/production further involves a general hierarchical physical structure which requires maintenance resources distribution at multiple sequential levels, and such hierarchical structure is beyond the scope of the mathematical model developed in Chapter 2. Therefore in Chapter 4 we built a more sophisticated new mathematical model following the maintenance approach from Chapter 2 in order to capture optimal maintenance planning decision making for more complex industrial problems that further involve the hierarchical structure.

Such research contribution, as specified in Chapter 4, is achieved through enlarging the mathematical modelling scope from the *production system-asset* structure to the *company-production system-asset* structure. Additionally, in such more complex modelling work, we provided an optional modelling choice for practitioners/researchers: approximately decompose the long-term part of maintenance decision making problem at the company level as a combination of sub-problems at the asset level. As discussed in Chapter 4, such modelling choice simplifies the resulted mathematical model and therefore reduces the total computational efforts required to solve the resulted mathematical problem.

Constructing a set of effective heuristics

As discussed in Chapter 2 and further illustrated in Chapter 4 and Chapter 5, the application of the maintenance approach from Chapter 2 to industrial cases result in large size mathematical problems which cannot be effectively solved by brute-force methods: we evaluate the effectiveness of different methods based on the numerical benchmarks on three criteria (these are computational time, data-storage cost and data accuracy level) which we specified in Section 2.4. Therefore we construct a set of heuristics in Chapter 4 in order to effectively solve such large size mathematical problems, including the problems which involve the aforementioned hierarchical structure.

As specified in Chapter 4, such set of heuristics consists of (1) the two-timescale decomposition method, (2) the parameters number bounding method, (3) Q-learning, (4) the polynomial function method and (5) the incremental gradient-descent method. Heuristic (1)-(2) are our own design and Heuristic (3)-(5) are selected from existing literature:

Heuristic (1) is the aforementioned optional modelling choice which reduces the size of both state space and action space of the resulted mathematical model and hence reduces the total computational efforts (in terms of computational time and data-storage cost) required to solve the mathematical problem; Heuristic (2) is used in combination with Heuristic (4), where the latter heuristic approximately stores the computational results by using polynomial functions (which is expected to be more effective compared to brute-force lookup tables in terms of data-storage cost), and the former heuristic further reduces the number of free parameters in each polynomial function to a known constant and such reduction effect results in additional improvement on computational time and data-storage cost; Heuristic (3) performs simulation-based approximate computation to solve the mathematical problem and such heuristic is expected to be more effective than brute-force computation methods in terms of computational time as discussed in Chapter 3; Heuristic (5) incrementally updates the free parameters in the aforementioned polynomial functions based on updated computational results from Heuristic (3): in other words, Heuristic (5) is a crucial component of the interface between aforementioned heuristics.

We numerically tested the heuristics on the power plant case study in simulation (see Chapter 5), and the simulation results empirically confirm the effectiveness of the heuristics.

Below we shall first in Section 6.1 discuss the main assumptions in our research work related to each such contribution, in order to provide a high level view about which assumptions are fundamental ones extracted from the problem setting in practice and which assumptions are chosen in this thesis to help us focus on the main features of the problem setting and simplify our modelling work: the latter type of assumptions is subject to potential changes in future studies; then we shall in Section 6.2 select from such change options and discuss further research directions.

## 6.1 Discussion on assumptions

### The performance-centred maintenance (PCM) approach

The PCM approach in general applies to industrial cases where the production system contains multiple individual assets working in parallel, given the set of assumptions (discussed in Chapter 2) embedded in the PCM approach are justifiable in the corresponding practical problem setting. Such assumptions are embedded in different aspects of the PCM approach: the business setting, the conceptual framework and the modelling framework; below we examine the main assumptions in each such aspect and discuss whether such assumptions are subject to changes in the future research.

Regarding the business setting, the main assumptions include the following: (1) the contracted period has a crucial impact on modifying the gap between the value perception perspectives of maintainers and operators, (2) no inventory (it is either impossible or very expensive to hold inventory) and (3) the output quantity of the system is determined by both the quantity and quality (for instance the coal grinding quality in the power plant case) of material processing of the individual assets. Assumption (1) is extracted from problem settings in practice (see specifications in Section 2.1) and it is fundamental to the PCM approach as such assumption underpins the development of the PCM approach in this thesis; therefore such assumption is unlikely to be changed in the future research. Assumption (2) is extracted from the power plant case but product storage may be a possible and reasonable choice in other industrial cases (such as in the car manufacturing industry); therefore Assumption (2) is subject to potential changes in the future research to suit a more general problem setting: **we shall in Section 6.2 discuss how the PCM approach should be updated given Assumption (2) is changed**. Assumption (3) is also extracted from the power plant case study and since it describes a more general situation compared to industrial cases where the output quantity of the system is solely determined by the quantity of material processing of the individual assets, the PCM approach can be applied to such cases.

Regarding the conceptual framework, the main assumptions include: (1) the maintainers understand the exact decision regime that the operators follow, (2) the machine state is continuously monitored and the monitoring information can accurately reveal the machine state, (3) all prices are all fixed throughout time, (4) the demand after the contracted period is independent and identical distributed (i.i.d.) and (5) discrete time-setting for both the machine deterioration process and operations decision making. Assumption (1) is extracted from problem settings in practice and it is fundamental to the PCM approach as such an assumption justifies building the machine utilisation behaviours of the operators into the maintainers' decision making in the mathematical model (see specifications in Section 2.2.1); therefore such assumption is unlikely to be changed in the future research. Assumptions (2)-(5) enable this research project to focus on the crucial features of fundamental trade-offs between managerial parties in their decision making, and such assumptions are all subject to potential changes in the future research in order to more accurately reflect practical problem settings: in reality, the machine state may only be checked at discrete time intervals and the monitoring information may not accurately reveal the machine state; prices fluctuate randomly over time; the future (non-contracted and hence non-observed) demand may not be well fitted by i.i.d. statistical models and the machine deterioration process is more likely to be time-continuous rather than time-discrete (note that the maintenance actions in practice usually take time lengths of several consecutive shifts, for example several days or weeks, and hence the discrete time-setting assumption for modelling decision making is less necessary to be changed in future studies). Regarding whether to change Assumption (4), practitioners/researchers can conduct model validation tests on historical demand data for i.i.d. statistical models of interest (so practitioners/researchers can estimate whether the future demand can be well fitted by i.i.d. statistical models) and/or conduct sensitivity analysis on the computational results (so practitioners/researchers can evaluate the impacts of potential statistical modelling inaccuracy on the quality of derived operations policies) in the context of their own case. For the cases where Assumption (4) should be changed, practitioners/researchers need to investigate more flexible modelling choices including Markov processes and time series forecasting. It is however worth emphasising that in practice the future (non-contracted) demand has less importance to operations decision making compared to already contracted demand, and we capture such difference in importance by introducing discount factor in our mathematical models; therefore we do not expect high necessity for changing Assumption (4) and we shall not further discuss such change option in this thesis. Instead, **we shall in Section 6.2 discuss how the PCM approach should be updated given the other three assumptions (Assumption (2), Assumption (3) and Assumption (5)) are changed**.

Regarding the modelling framework, the main assumptions include: (1) the system state evolving process satisfies the Markovian property, (2) the transition between any system states always consumes a fixed time length and (3) the performance and condition of every asset can each be perceived as a single-dimensional aggregated variable which can be further categorised into ordered discrete levels. Such assumptions enable us to model the deterioration and maintenance process by using a discrete-time discrete-state Markov decision process. The three assumptions are not necessarily accurate reflection of practical problem settings: **Assumption (2) can be changed to better suit time-continuous deterioration processes in practice, which we shall further discuss in Section 6.2**; Assumption (3) aligns with a common approximation perception of the machine state from engineers in practice but it is worth knowing that such aggregation perception may comprise the quality of derived operations policies if each asset in practice further consists of multiple non-identical components each of which has a different deterioration mechanism: in such case it is necessary to consider **changing Assumption (3) and conducting more complex modelling work in order to better capture asset performance deterioration and/or asset condition deterioration on multiple**

**dimensions, which we shall further discuss in Section 6.2**; regarding Assumption (1), it is unclear what modelling frameworks can replace Markov decision processes while maintaining the same level of modelling simplicity if Assumption (1) is changed.

**The mathematical model for the hierarchical structure**

The hierarchical structure of the machine systems significantly increases the complexity of decision making, rendering the mathematical model much more complex. See comparison between the three-stage hybrid MDP model in Section 4.2 and the mathematical model in Section 2.2.4.3.

The three-stage hybrid MDP model is developed in the context of the power plant case, where all the mills are identical in terms of their physical structure. However, we would like to emphasise that Stage (1) and Stage (2) of the model does not require the assets to be identical; as for Stage (3) of the mathematical model which is resulted from the application of the two-timescale decomposition method (see Section 4.2.1.1), both the decomposition method and the mathematical model part can be relatively easily adapted for industrial cases in which the assets are not identical, as discussed in Section 4.2.1.1.

Additionally, the three-stage hybrid MDP model follows the specific maintenance resources constraints in the power plant case study: the plant has two maintenance crews in total: one dedicates to service and the other crew dedicates to overhaul. The mathematical model can be relatively easily adapted for other industrial cases which has either more or less maintenance crews: to suit such cases, the action space in Stage (1) and Stage (2) of the mathematical model should be expanded or reduced accordingly, and in Stage (3) the maintenance crew rotation schedules should also be modified to reflect the change of crew numbers; these are relatively simple modifications and we shall not further discuss.

**The heuristics**

The set of heuristics is also developed in the context of the power plant case, and such set of heuristics can be relatively easily adapted for other industrial cases of interest as discussed in Section 4.2-4.3; here we focus on assumptions embedded in the polynomial function method and the parameters number bounding method (the two-timescale decomposition method is discussed above already).

The choice of polynomial function method is justified by the extrapolated concave data patterns in the case study (see discussion in Section 4.3.1); furthermore, we choose to use *quadratic* functions to approximate the expected value of interest, based on numerical tests on medium size problems of the case study. Although we assume similar concave data patters in general also exist in other industrial cases of interest and therefore polynomial function method can also be applied to such cases, *quadratic* functions may not yield relative accurate approximation in such cases and hence polynomial functions with higher degrees may be required (for instance cubic functions): note that polynomial functions with higher degrees still contain concave properties. Higher degree polynomial functions however may introduce a large number of free parameters to be tuned; additionally, the specific quadratic function setting for the power plant case study does not involve cross products of variables from different units in the power plant (see specifications in Section 4.3.1): such modelling choice is justified by the specific problem setting where each unit in the plant has its separate supply contract (see discussion in Section 4.3.1), but other industrial cases may not have such convenient problem feature and hence an extra large number of free parameters (for the cross products) may be required to be tuned. Therefore, the parameters number bounding method is crucial (even for quadratic functions) to ensure the total number of free parameters is relatively small.

The parameters number bounding method has a fundamental prerequisite that all assets are identical, in order to guarantee the value approximation quality (of the polynomial function method) is not compromised due to the application of such method; but the prerequisite may not be satisfied in some industrial cases, and in such cases the application of the parameters number bounding method may instead largely compromise the value approximation quality. It is however unclear how to further improve the method to mitigate such impacts on data accuracy if the prerequisite is changed.

## 6.2 Further research

Section 6.1 discusses which assumptions in our research work are subject to potential changes; here we choose to focus on the change options that would bring major impacts to our research and discuss the implied updates in the modelling work for future research.

- Assumption change option (1): keeping inventory is a potentially reasonable choice.The choice of keeping inventory would enable the operators to exploit excessive production capacity of the machine system in practice; in order to facilitate optimal inventory decision making on buffer size(s), the PCM approach needs to consider additional trade-offs involved in such decision making from a balanced view between machine utilisation and increased risk of failure: exploiting extra production helps reduce the necessity of speeding up future production in the contracted period and helps cut down potential penalty cost; however, inventory cost must be evaluated and additionally the reduction of maintenance opportunities (a machine cannot produce products and receive maintenance at the same time) should be examined. Meanwhile, the operators may be motivated to expand the production planning-horizon beyond the contracted period, as the inventory level at the end of the current contracted period directly impacts whether the demand to be contracted in the future can be fulfilled by self-production in the company. The decision making on buffer size(s), as well as the additional trade-offs and impacts on production planning-horizon, should be captured in the conceptual framework and the mathematical model of the PCM approach if assumption change option (1) is chosen.

- Assumption change option (2): the relevant prices evolve stochastically over time. Such assumption aligns with practice: in reality, the raw material purchase price, the product sales price and the emergent procurement price on spot market all fluctuate randomly over time; therefore in practice the penalty cost and the profit per unit of demand discussed in the business setting of the PCM approach would also randomly fluctuate over time. Intuitively speaking, as the unit profit or penalty cost becomes larger, production should be more likely prioritised over maintenance in practice (meaning maintenance should be more likely scheduled for a later time) and speeded-up production should be more likely chosen to resolve potential production shortfalls (to avoid high penalty cost) for better costs/benefits; in other words, the price dynamics have important impacts on operations decision making. Given assumption change option (2), such impacts in practice should be captured at each level of the PCM approach: the conceptual framework, the modelling framework and the mathematical model. Such research direction relates to an active research area which focuses on the interface between the commodity markets and operations management where the operations decision making is subject to the randomly fluctuating prices on different commodity markets (for example, forward market and spot market where procurement/sales is agreed on forward contracts and spot contracts respectively); the studies (e.g. [47, 62, 63, 100]) in such research area provide example modelling frameworks that incorporate stochastically evolving prices into operations decision making, which we may further investigate.

- Assumption change option (3): the assets deteriorate in a continuous time-setting. As a result, the modelling framework and the mathematical model of the PCM approach should be updated to suit the more general problem settings that the assumption change option reflects. More specifically speaking, semi-Markov process or continuous-time Markov process (see discussion of both frameworks in Section 1.1.2.2) can be adopted in the PCM approach to replace Markov chain to model the machine deterioration process.

- Assumption change option (4): the performance and condition of an asset can each potentially deteriorates on multiple dimensions, and additionally the deterioration mechanisms can be highly different between such dimensions. As a result, the deterioration measurement on each such dimension should be modelled as a separate entity in the mathematical model, and additionally the transition probabilities elicitation framework specified in Section 2.2.3 should be updated to elicit the machine state transition probabilities on each aforementioned dimension separately (assuming the measurement on each such dimension can be categorised into ordered discrete levels) rather than at an aggregated level. However, such more complex modelling work induces exponential increment on the state space size in the resulted mathematical problem, and such exponential increment inevitably requires more effective computation and data-storage heuristics which raises further interesting research questions. We believe that as machines are becoming more complex in the manufacturing industry, such a more complex modelling framework and the designing of more effective heuristics are worth further investigation.

- Assumption change option (5): the machine monitoring information can only reveal part of the machine state. So far in practice, it is common that the machine state cannot be fully monitored, especially if the machine has a complex physical structure. In order to capture such more general problem setting where operations decision making relies on incomplete machine state information, assumption change option (5) can be chosen and consequently the modelling framework and the mathematical model of the PCM approach should be updated to reflect such additional uncertainty faced by decision makers. More specifically speaking, hidden Markov process (discussed in Section 1.1.2.2) can be adopted in the PCM approach to replace Markov chain to model the machine deterioration process. However, as modern-day monitoring technologies advance in the current era of Industry 4.0 and machine monitoring becomes more effective (especially with increasing use of advanced sensors), the issue of inaccurate monitoring would become less pressing and therefore the necessity of mathematically capturing the uncertainty involved in machine state monitoring may diminish.

Nevertheless, while the technology of ubiquitous sensing is materialising and hence decision makers are enabled to harvest more accurate and comprehensive monitoring information, researchers/practitioners face an ever-growing challenge of balancing between model complexity and model accuracy: in order to *fully* capture the more detailed monitoring information in the mathematical model, the machine state may need to be mathematically described in very complex ways (for example described as a high-dimensional variable) and hence the resulted mathematical model may become very complex and even computational intractable; therefore, rather than blindly pursuing a "true" mathematical model which *fully* captures the monitoring information in the ever-more ubiquitous sensing industrial setting, researchers and practitioners should find a suitable simplified mathematical description of the machine state which provides a useful approximation.

Of course, the advancement of computation technologies (for example the breakthroughs in quantum computing) would tilt the balance and enable researchers/practitioners to pursue more accurate models without losing computational tractability; however, the trade-off between model complexity and model accuracy would always remain and therefore reaching a balance in between (rather than building an absolutely accurate model that involves over-elaboration and over-parameterization) is always the key of a good mathematical modelling work. As summarised by [20], *all models are wrong but some are useful.*

Given the five different future research directions discussed above, we choose to first look into assumption change option (3) in our immediate next study, as such choice induces perhaps the lowest level of extra modelling complexity to our existing work; the other four research directions shall be investigated in later studies.

# Bibliography

[1] Aggoune, R. [2004]. Minimizing the makespan for the flow shop scheduling problem with availability constraints, *European Journal of Operational Research* **153**(3): 534–543.

[2] Ahmad, R. and Kamaruddin, S. [2012]. An overview of time-based and condition-based maintenance in industrial application, *Computers & Industrial Engineering* **63**(1): 135–149.

[3] Alaswad, S. and Xiang, Y. [2017]. A review on condition-based maintenance optimization models for stochastically deteriorating system, *Reliability Engineering & System Safety* **157**: 54–63.

[4] Alkali, B. M., Bedford, T., Quigley, J. and Gaw, J. [2009]. Failure and maintenance data extraction from power plant maintenance management databases, *Journal of Statistical Planning and Inference* **139**(5): 1766–1776.

[5] Alle, A., Papageorgiou, L. G. and Pinto, J. M. [2004]. A mathematical programming approach for cyclic production and cleaning scheduling of multistage continuous plants, *Computers & Chemical Engineering* **28**(1): 3–15.

[6] Alle, A., Pinto, J. M. and Papageorgiou, L. G. [2004]. The economic lot scheduling problem under performance decay, *Industrial & Engineering Chemistry Research* **43**(20): 6463–6475.

[7] Barlow, E., Revie, M., Bedford, T. and Walls, L. [2013]. Trading off asset performance and condition to model strategic maintenance decisions, *In: Safety, Reliability and Risk Analysis* pp. 723–731.

[8] Barlow, E., Revie, M., Bedford, T., Walls, L. and Junchi, T. [2017]. The performance-centred approach to optimising the maintenance of a complex system. Unpublished.

[9] Barlow, R. and Hunter, L. [1960]. Optimum preventive maintenance policies, *Operations Research* **8**(1): 90–100.

[10] Beichelt, F. [1982]. A replacement policy based on limits for the repair cost rate, *IEEE Transactions on Reliability* **31**(4): 401–403.

[11] Bellman, R. [1957]. A markovian decision process, *Journal of Mathematics and Mechanics* **6**: 679–684.

[12] Bellman, R. E. and Dreyfus, S. E. [1962]. *Applied dynamic programming*, Princeton university press.

[13] Bertsekas, D. P. [1995]. *Dynamic programming and optimal control*, Vol. 1, Athena scientific Belmont, MA.

[14] Bi, R., Zhou, C. and Hepburn, D. M. [2017]. Detection and classification of faults in pitch-regulated wind turbine generators using normal behaviour models based on performance curves, *Renewable Energy* **105**: 674–688.

[15] Biondi, M., Sand, G. and Harjunkoski, I. [2017]. Optimization of multipurpose process plant operations: A multi-time-scale maintenance and production scheduling approach, *Computers & Chemical Engineering* **99**: 325–339.

[16] Bleakie, A. and Djurdjanovic, D. [2013]. Feature extraction, condition monitoring, and fault modeling in semiconductor manufacturing systems, *Computers in Industry* **64**(3): 203–213.

[17] Blischke, W. R. and Murthy, D. N. P. [2000]. *Reliability: modeling, prediction, and optimization*, Wiley.

[18] Bock, S., Briskorn, D. and Horbach, A. [2012]. Scheduling flexible maintenance activities subject to job-dependent machine deterioration, *Journal of Scheduling* **15**(5): 565–578.

[19] Bot, Y. and Azoulay, D. [2015]. Asset maintenance simulation: The case-study of an offshore wind farm, *Reliability and Maintainability Symposium, 2015 Annual*, IEEE, pp. 1–6.

[20] Box, G. E. and Draper, N. R. [1987]. *Empirical model-building and response surfaces*, John Wiley & Sons.

[21] Broomhead, D. S. and Lowe, D. [n.d.]. Multi-variable functional interpolation and adaptive networks, *Complex Systems* **2**: 321–355.

[22] Buhmann, M. D. [2003]. *Radial basis functions: theory and implementations*, Vol. 12, Cambridge university press.

[23] Cai, Y., Hasenbein, J. J., Kutanoglu, E. and Liao, M. [2013]. Single-machine multiple-recipe predictive maintenance, *Probability in the Engineering and Informational Sciences* **27**(2): 209–235.

[24] Campos, J. [2009]. Development in the application of ict in condition monitoring and maintenance, *Computers in Industry* **60**(1): 1–20.

[25] Cartella, F., Lemeire, J., Dimiccoli, L. and Sahli, H. [2015]. Hidden semi-markov models for predictive maintenance, *Mathematical Problems in Engineering* **2015**: 1–23.

[26] Casas-Liza, J., Pinto, J. and Papageorgiou, L. [2005]. Mixed integer optimization for cyclic scheduling of multiproduct plants under exponential performance decay, *Chemical Engineering Research and Design* **83**(10): 1208–1217.

[27] Castro, I. T. and Mercier, S. [2016]. Performance measures for a deteriorating system subject to imperfect maintenance and delayed repairs, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* **230**(4): 364–377.

[28] Castro, P. M., Grossmann, I. E., Veldhuizen, P. and Esplin, D. [2014]. Optimal maintenance scheduling of a gas engine power plant using generalized disjunctive programming, *American Institute of Chemical Engineers* **60**(6): 2083–2097.

[29] Chen, D. and Trivedi, K. S. [2002]. Closed-form analytical results for condition-based maintenance, *Reliability Engineering & System Safety* **76**(1): 43–51.

[30] Chen, Y.-L., Chang, C.-C. and Sheu, D.-F. [2016]. Optimum random and age replacement policies for customer-demand multi-state system reliability under imperfect maintenance, *International Journal of Systems Science* **47**(5): 1130–1141.

[31] Cheung, K.-Y., Hui, C.-W., Sakamoto, H., Hirata, K. and O'Young, L. [2004]. Short-term site-wide maintenance scheduling, *Computers & Chemical Engineering* **28**(1): 91–102.

[32] Çimen, M. and Kirkbride, C. [2017]. Approximate dynamic programming algorithms for multidimensional flexible production-inventory problems, *International Journal of Production Research* **55**(7): 2034–2050.

[33] Cybenko, G. [1989]. Approximation by superpositions of a sigmoidal function, *Mathematics of Control, Signals, and Systems* **2**(4): 303–314.

[34] Dai, P. [2007]. *Faster dynamic programming for markov decision processes*, Master's thesis, University of Kentucky.

[35] Dai, P. and Goldsmith, J. [2007a]. Multi-threaded blao* algorithm, *In Proc. 20th International FLAIRS Conference*, pp. 56–62.

[36] Dai, P. and Goldsmith, J. [2007b]. Topological value iteration algorithm for markov decision processes, *International Joint Conference on Artificial Intelligence*, pp. 1860–1865.

[37] Dai, P. and Weld, D. S. [2009]. Focused topological value iteration, *International Conference on Automated Planning and Scheduling*.

[38] Dai, P., Weld, D. S. and Goldsmith, J. [2011]. Topological value iteration algorithms, *Journal of Artificial Intelligence Research* **42**: 181–209.

[39] Darken, C., Chang, J. and Moody, J. [1992]. Learning rate schedules for faster stochastic gradient search, *Neural Networks for Signal Processing [1992] II*, IEEE, pp. 3–12.

[40] Das, T. K., Gosavi, A., Mahadevan, S. and Marchalleck, N. [1999]. Solving semi-markov decision problems using average reward reinforcement learning, *Management Science* **45**(4): 560–574.

[41] Das, T. K. and Sarkar, S. [1999]. Optimal preventive maintenance in a production inventory system, *IIE Transactions* **31**(6): 537–551.

[42] Dasarathy, B. V. [1991]. *Nearest neighbor (NN) norms:NN pattern classification techniques*, McGraw-Hill Computer Science Series. IEEE Computer Society Press, Las Alamitos, California.

[43] David, C. R. et al. [1972]. Regression models and life tables (with discussion), *Journal of the Royal Statistical Society. Series B (Methodological)* **34**: 187–220.

[44] de Jonge, B., Teunter, R. and Tinga, T. [2017]. The influence of practical factors on the benefits of condition-based maintenance over time-based maintenance, *Reliability Engineering & System Safety* **158**: 21–30.

[45] Dedopoulos, I. T. and Shah, N. [1995]. Optimal short-term scheduling of maintenance and production for multipurpose plants, *Industrial & Engineering Chemistry Research* **34**(1): 192–201.

[46] Deloux, E., Castanier, B. and Bérenguer, C. [2009]. Predictive maintenance policy for a gradually deteriorating system subject to stress, *Reliability Engineering & System Safety* **94**(2): 418–431.

[47] Devalkar, S. K., Anupindi, R. and Sinha, A. [2017]. Dynamic risk management of commodity operations: Model and analysis, *Manufacturing & Service Operations Management* .
**URL:** *https://doi.org/10.1287/msom.2017.0647*

[48] Deza, M. M. and Deza, E. [2009]. *Encyclopedia of distances*, Springer.

[49] Dhillon, B. S. [2002]. *Engineering maintenance: a modern approach*, CRC Press.

[50] Ding, S.-H. and Kamaruddin, S. [2015]. Maintenance policy optimization-literature review and directions, *The International Journal of Advanced Manufacturing Technology* **76**(5-8): 1263–1283.

[51] Dohi, T., Matsushima, N., Kaio, N. and Osaki, S. [1997]. Nonparametric repair-limit replacement policies with imperfect repair, *European Journal of Operational Research* **96**(2): 260–273.

[52] Drinkwater, R. and Hastings, N. A. [1967]. An economic replacement model, *Operations Research* **18**(2): 121–138.

[53] Duffuaa, S., Ben-Daya, M., Al-Sultan, K. and Andijani, A. [2001]. A generic conceptual simulation model for maintenance systems, *Journal of Quality in Maintenance Engineering* **7**(3): 207–219.

[54] El-Gohary, A. [2004]. Estimations of parameters in a three state reliability semi-markov model, *Applied Mathematics and Computation* **154**(2): 389–403.

[55] Emami-Mehrgani, B., Nadeau, S. and Kenné, J.-P. [2014]. Optimal lockout/tagout, preventive maintenance, human error and production policies of manufacturing systems with passive redundancy, *Journal of Quality in Maintenance Engineering* **20**(4): 453–470.

[56] Emigh, M. S., Kriminger, E. G., Brockmeier, A. J., Príncipe, J. C. and Pardalos, P. M. [2016]. Reinforcement learning in video games using nearest neighbor interpolation and metric learning, *IEEE Transactions on Computational Intelligence and AI in Games* **8**(1): 56–66.

[57] Fan, L., Jiang, C. and Hu, M. [2017]. Ground track maintenance for beidou igso satellites subject to tesseral resonances and the luni-solar perturbations, *Advances in Space Research* **59**(3): 753–761.

[58] Gao, Y., Xie, N., Hu, K., Zhu, Y. and Wang, L. [2017]. An optimized clustering approach using simulated annealing algorithm with hmm coordination for rolling elements bearings' diagnosis, *Journal of Failure Analysis and Prevention* **17**(3): 1–18.

[59] Georgiadis, M. C. and Papageorgiou, L. G. [2001]. Optimal scheduling of heat-integrated multipurpose plants under fouling conditions, *Applied Thermal Engineering* **21**(16): 1675–1697.

[60] Georgiadis, M. and Papageorgiou, L. [2000]. Optimal energy and cleaning manage-

ment in heat exchanger networks under fouling, *Chemical Engineering Research and Design* **78**(2): 168–179.

[61] Gertsbakh, I. [2013]. *Reliability theory: with applications to preventive maintenance*, Springer.

[62] Goel, A. and Gutierrez, G. J. [2011]. Multiechelon procurement and distribution policies for traded commodities, *Management Science* **57**(12): 2228–2244.

[63] Goel, A. and Tanrısever, F. [2013]. Financial hedging and optimal procurement policies under correlated price and demand, *Production and Operations Management* **26**(10): 1924–1945.

[64] Gosavi, A. [2003]. *Simulation-based optimization*, Springer.

[65] Gosavi, A. [2004a]. A reinforcement learning algorithm based on policy iteration for average reward: Empirical results with yield management and convergence analysis, *Machine Learning* **55**(1): 5–29.

[66] Gosavi, A. [2004b]. Reinforcement learning for long-run average cost, *European Journal of Operational Research* **155**(3): 654–674.

[67] Gosavi, A. [2006]. A risk-sensitive approach to total productive maintenance, *Automatica* **42**(8): 1321–1330.

[68] Gosavi, A. [2008]. On step sizes, stochastic shortest paths, and survival probabilities in reinforcement learning, *Simulation Conference, 2008. WSC 2008. Winter*, IEEE, pp. 525–531.

[69] Gosavi, A. [2009]. Reinforcement learning: A tutorial survey and recent advances, *INFORMS Journal on Computing* **21**(2): 178–192.

[70] Gosavi, A. [2010]. Finite horizon markov control with one-step variance penalties, *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference*, IEEE, pp. 1355–1359.

[71] Gosavi, A. [2014]. Variance-penalized markov decision processes: Dynamic programming and reinforcement learning techniques, *International Journal of General Systems* **43**(6): 649–669.

[72] GOSAVII, A., Bandla, N. and Das, T. K. [2002]. A reinforcement learning approach to a single leg airline revenue management problem with multiple fare classes and overbooking, *IIE Transactions* **34**(9): 729–742.

[73] Grall, A., Dieulle, L., Bérenguer, C. and Roussignol, M. [2002]. Continuous-time predictive-maintenance scheduling for a deteriorating system, *IEEE Transactions on Reliability* **51**(2): 141–150.

[74] Gürler, Ü. and Kaya, A. [2002]. A maintenance policy for a system with multi-state components: an approximate solution, *Reliability Engineering & System Safety* **76**(2): 117–127.

[75] Harper, M., Knight, V., Jones, M., Koutsovoulos, G., Glynatsi, N. E. and Campbell, O. [2017]. Reinforcement learning produces dominant strategies for the iterated prisoner's dilemma, *PloS one* **12**(12): e0188046.

[76] Hazaras, M. J., Swartz, C. L. and Marlin, T. E. [2012]. Flexible maintenance within a continuous-time state-task network framework, *Computers & Chemical Engineering* **46**: 167–177.

[77] Heluane, H., Blanco, A. M., Hernández, M. R. and Bandoni, J. A. [2012]. Simultaneous re-design and scheduling of multiple effect evaporator systems, *Computers & Operations Research* **39**(5): 1173–1186.

[78] Heluane, H., Colombo, M. A., Hernandez, M. R., Sequei ra, S. E., GRAELLS, M. and Puigjaner, L. [2004]. Scheduling of continuous parallel lines in the evaporation section of sugar plants, *Chemical Engineering Communications* **191**(9): 1121–1146.

[79] Heluane, H., Colombo, M., Hernández, M., Graells, M. and Puigjaner, L. [2007]. Enhancing sugar cane process performance through optimal production scheduling, *Chemical Engineering and Processing: Process Intensification* **46**(3): 198–209.

[80] Hernández-Lerma, O. and Lasserre, J. B. [2012]. *Further topics on discrete-time Markov control processes*, Vol. 42, Springer Science & Business Media.

[81] Hontelez, J. A., Burger, H. H. and Wijnmalen, D. J. [1996]. Optimum condition-based maintenance policies for deteriorating systems with partial information, *Reliability Engineering & System Safety* **51**(3): 267–274.

[82] Howard, R. A. [1960]. *Dynamic Programming and Markov Processes*, MIT Press.

[83] Hutter, M. [2016]. Extreme state aggregation beyond markov decision processes, *Theoretical Computer Science* **650**: 73–91.

[84] Huynh, K. T., Barros, A., Berenguer, C. and Castro, I. T. [2011]. A periodic inspection and replacement policy for systems subject to competing failure modes due to degradation and traumatic events, *Reliability Engineering & System Safety* **96**(4): 497–508.

[85] Jain, V. and Grossmann, I. E. [1998]. Cyclic scheduling of continuous parallel-process units with decaying performance, *American Institute of Chemical Engineers* **44**(7): 1623–1636.

[86] Jardine, A. K., Lin, D. and Banjevic, D. [2006]. A review on machinery diagnostics and prognostics implementing condition-based maintenance, *Mechanical Systems and Signal Processing* **20**(7): 1483–1510.

[87] Kapur, P., Garg, R. and Butani, N. [1989]. Some replacement policies with minimal repairs and repair cost limit, *International Journal of Systems Science* **20**(2): 267–279.

[88] Karamatsoukis, C. and Kyriakidis, E. [2010]. Optimal maintenance of two stochastically deteriorating machines with an intermediate buffer, *European Journal of Operational Research* **207**(1): 297–308.

[89] Kenné, J. P. and Nkeungoue, L. [2008]. Simultaneous control of production, preventive and corrective maintenance rates of a failure-prone manufacturing system, *Applied Numerical Mathematics* **58**(2): 180–194.

[90] Kijima, M., Morimura, H. and Suzuki, Y. [1988]. Periodical replacement problem without assuming minimal repair, *European Journal of Operational Research* **37**(2): 194–203.

[91] Kijima, M. and Nakagawa, T. [1992]. Replacement policies of a shock model with imperfect preventive maintenance, *European Journal of Operational Research* **57**(1): 100–110.

[92] Kobbacy, K. A. H. and Murthy, D. P. [2008]. *Complex system maintenance handbook*, Springer Science & Business Media.

[93] Koutras, V., Malefaki, S. and Platis, A. [2017]. Optimization of the dependability and performance measures of a generic model for multi-state deteriorating systems under maintenance, *Reliability Engineering & System Safety* **166**: 73–86.

[94] Lie, C. H. and Chun, Y. H. [1986]. An algorithm for preventive maintenance policy, *IEEE Transactions on Reliability* **35**(1): 71–75.

[95] Liu, S., Yahia, A. and Papageorgiou, L. G. [2014]. Optimal production and mainten-

ance planning of biopharmaceutical manufacturing under performance decay, *Industrial & Engineering Chemistry Research* **53**(44): 17075–17091.

[96] Liu, T., Chen, J. and Dong, G. [2014]. Zero crossing and coupled hidden markov model for a rolling bearing performance degradation assessment, *Journal of Vibration and Control* **20**(16): 2487–2500.

[97] Love, C. and Guo, R. [1996]. Utilizing weibull failure rates in repair limit analysis for equipment replacement/preventive maintenance decisions, *Journal of the Operational Research Society* **47**(11): 1366–1376.

[98] Love, C., Zhang, Z. G., Zitron, M. and Guo, R. [2000]. A discrete semi-markov decision model to determine the optimal repair/replacement policy under general repairs, *European Journal of Operational Research* **125**(2): 398–409.

[99] Madu, C. N. [2000]. Competing through maintenance strategies, *International Journal of Quality & Reliability Management* **17**(9): 937–949.

[100] Mahapatra, S., Levental, S. and Narasimhan, R. [2017]. Market price uncertainty, risk aversion and procurement: Combining contracts and open market sourcing alternatives, *International Journal of Production Economics* **185**: 34–51.

[101] Makabe, H. and Morimura, H. [1963]. A new policy for preventive maintenance, *Journal of Operations Research Society of Japan* **5**: 110–124.

[102] Manatos, A., Koutras, V. P. and Platis, A. N. [2016]. Dependability and performance stochastic modelling of a two-unit repairable production system with preventive maintenance, *International Journal of Production Research* **54**(21): 6395–6415.

[103] Mann, L., Saxena, A. and Knapp, G. M. [1995]. Statistical-based or condition-based preventive maintenance?, *Journal of Quality in Maintenance Engineering* **1**(1): 46–59.

[104] Mobley, R. K. [2002]. *An introduction to predictive maintenance*, Butterworth-Heinemann.

[105] Morimura, H. [1969]. *On some preventive maintenance policies for IFR*, University of North Carolina, Dept. of Statistics.

[106] Nakagawa, T. [1984]. Optimal policy of continuous and discrete replacement with minimal repair at failure, *Naval Research Logistics* **31**(4): 543–550.

[107] Nakagawa, T. [1986]. Periodic and sequential preventive maintenance policies, *Journal of Applied Probability* **23**(2): 536–542.

[108] Nakagawa, T. and Osaki, S. [1974]. The optimum repair limit replacement policies, *Operational Research Quarterly* **25**(2): 311–317.

[109] Natvig, B. [2010]. *Multistate systems reliability theory with applications*, John Wiley & Sons.

[110] Nguyen, D. and Murthy, D. [1981]. Optimal preventive maintenance policies for repairable systems, *Operations Research* **29**(6): 1181–1194.

[111] Nie, Y., Biegler, L. T., Wassick, J. M. and Villa, C. M. [2014]. Extended discrete-time resource task network formulation for the reactive scheduling of a mixed batch/continuous process, *Industrial & Engineering Chemistry Research* **53**(44): 17112–17123.

[112] 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.

[113] OConnor, P. D. T. and Kleyner, A. [2015]. *Practical reliability engineering*, Wiley.

[114] Padakandla, S., Prabuchandran, K. and Bhatnagar, S. [2015]. Energy sharing for multiple sensor nodes with finite buffers, *IEEE Transactions on Communications* **63**(5): 1811–1823.

[115] Papageorgiou, G. and Mouratidis, A. [2015]. Defining threshold values for pavement surface characteristics, *Proceedings of the Institution of Civil Engineers-Transport* **168**(3): 223–230.

[116] Petrik, M. and Subramanian, D. [2014]. Raam: The benefits of robustness in approximating aggregated mdps in reinforcement learning, *Advances in Neural Information Processing Systems 27*, pp. 1979–1987.

[117] Pham, H. and Wang, H. [1996]. Imperfect maintenance, *European Journal of Operational Research* **94**(3): 425–438.

[118] Powell, W. B. [2007]. *Approximate Dynamic Programming: Solving the curses of dimensionality*, Vol. 703, John Wiley & Sons.

[119] Rahaman, Z. and Sil, J. [2014]. De based q-learning algorithm to improve speed of convergence in large search space applications, *International Conference on Electronic Systems*, IEEE, pp. 408–412.

[120] Rao, P. N. S. and Naikan, V. A. [2006]. A condition-based preventive maintenance policy for markov deteriorating systems, *International Journal of Performability Engineering* **2**(2): 175.

[121] Read, J., Žliobaitė, I. and Hollmén, J. [2016]. Labeling sensing data for mobility modeling, *Information Systems* **57**: 207–222.

[122] Ross, S. M. [1983]. *Introduction to stochastic dynamic programming*, Academic press.

[123] Rummery, G. A. and Niranjan, M. [1994]. *On-line Q-learning using connectionist systems*, Vol. 37, University of Cambridge, Department of Engineering.

[124] Samuelson, A., Haigh, A., O'Reilly, M. M. and Bean, N. G. [2016]. Stochastic model for maintenance in continuously deteriorating systems, *European Journal of Operational Research* **259**(3): 1169–1179.

[125] Schulz, E. P., Bandoni, J. A. and Diaz, M. S. [2006]. Optimal shutdown policy for maintenance of cracking furnaces in ethylene plants, *Industrial & Engineering Chemistry Research* **45**(8): 2748–2757.

[126] Shafiee, M., Finkelstein, M. and Bérenguer, C. [2015]. An opportunistic condition-based maintenance policy for offshore wind turbine blades subjected to degradation and environmental shocks, *Reliability Engineering & System Safety* **142**: 463–471.

[127] Sheu, S.-H., Chang, C.-C., Chen, Y.-L. and Zhang, Z. G. [2015]. Optimal preventive maintenance and repair policies for multi-state systems, *Reliability Engineering & System Safety* **140**: 78–87.

[128] Shin, J.-H. and Jun, H.-B. [2015]. On condition based maintenance policy, *Journal of Computational Design and Engineering* **2**(2): 119–127.

[129] Shone, R. W. [2014]. *Optimal control of queueing systems with multiple heterogeneous facilities*, PhD thesis, Cardiff University.

[130] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V. and Lanctot, M. [2016]. Mastering the game of go with deep neural networks and tree search, *Nature* **529**(7587): 484–489.

[131] Stefán, P., Monostori, L. and Erdélyi, F. [2001]. Reinforcement learning for solving

shortest-path and dynamic scheduling problems, *Proceedings of the 3rd International Workshop on Emergent Synthesis, IWES*, Vol. 1, pp. 83–88.

[132] Sutton, R. S. [1988]. Learning to predict by the methods of temporal differences, *Machine learning* **3**(1): 9–44.

[133] Sutton, R. S. and Barto, A. G. [1998]. *Reinforcement learning: An introduction*, MIT Press.

[134] Tadepalli, P. and Ok, D. [1998]. Model-based average reward reinforcement learning, *Artificial Intelligence* **100**(1): 177–224.

[135] *The GB electricity wholesale market* [2015].
**URL:** *https://www.ofgem.gov.uk/electricity/wholesale-market/gb-electricity-wholesale-market*

[136] Tipaldi, M. and Glielmo, L. [2016]. State aggregation approximate dynamic programming for model-based spacecraft autonomy, *European Control Conference, 2016 European*, IEEE, pp. 86–91.

[137] Tsang, A. H. [1995]. Condition-based maintenance: tools and decision making, *Journal of Quality in Maintenance Engineering* **1**(3): 3–17.

[138] Uğurlu, K. [2017]. Controlled markov decision processes with avar criteria for unbounded costs, *Journal of Computational and Applied Mathematics* **319**: 24–37.

[139] Upton, A., Jefferson, B., Moore, G. and Jarvis, P. [2017]. Rapid gravity filtration operational performance assessment and diagnosis for preventative maintenance from on-line data, *Chemical Engineering Journal* **313**: 250–260.

[140] Usher, J. S., Kamal, A. H. and Syed, W. H. [1998]. Cost optimal preventive maintenance and replacement scheduling, *IIE Transactions* **30**(12): 1121–1128.

[141] van der Weide, J. A. and Pandey, M. D. [2011]. Stochastic analysis of shock process and modeling of condition-based maintenance, *Reliability Engineering & System Safety* **96**(6): 619–626.

[142] van der Weide, J. A., Pandey, M. D. and van Noortwijk, J. M. [2010]. Discounted cost model for condition-based maintenance optimization, *Reliability Engineering & System Safety* **95**(3): 236–246.

[143] Van Roy, B., Bertsekas, D. P., Lee, Y. and Tsitsiklis, J. N. [1997]. A neuro-dynamic programming approach to retailer inventory management, *The 36th IEEE Conference on Decision and Control*, Vol. 4, IEEE, pp. 4052–4057.

[144] Varnier, C. and Zerhouni, N. [2012]. Scheduling predictive maintenance in flow-shop, *Prognostics and System Health Management (PHM)*, IEEE, pp. 1–6.

[145] Wang, H. [2002]. A survey of maintenance policies of deteriorating systems, *European Journal of Operational Research* **139**(3): 469–489.

[146] Wang, H. and Pham, H. [1999]. Some maintenance models and availability withimperfect maintenance in production systems, *Annals of Operations Research* **91**: 305–318.

[147] Wang, L., Chu, J. and Mao, W. [2009]. A condition-based replacement and spare provisioning policy for deteriorating systems with uncertain deterioration to failure, *European Journal of Operational Research* **194**(1): 184–205.

[148] Watkins, C. J. C. H. [1989]. *Learning from delayed rewards*, PhD thesis, King's College, Cambridge.

[149] Watkins, C. J. and Dayan, P. [1992]. Q-learning, *Machine learning* **8**(3-4): 279–292.

[150] Whitmore, G. and Schenkelberg, F. [1997]. Modelling accelerated degradation data using wiener diffusion with a time scale transformation, *Lifetime Data Analysis* **3**(1): 27–45.

[151] Xia, L., Zhao, Q. and Jia, Q.-S. [2008]. A structure property of optimal policies for maintenance problems withsafety-critical components, *IEEE Transactions on Automation Science and Engineering* **5**(3): 519–531.

[152] Xiang, Y., Cassady, C. R., Jin, T. and Zhang, C. W. [2014]. Joint production and maintenance planning with machine deterioration and random yield, *International Journal of Production Research* **52**(6): 1644–1657.

[153] Ye, Z.-S. and Chen, N. [2014]. The inverse gaussian process as a degradation model, *Technometrics* **56**(3): 302–311.

[154] Yu, J. [2017]. Adaptive hidden markov model-based online learning framework for bearing faulty detection and performance degradation monitoring, *Mechanical Systems and Signal Processing* **83**: 149–162.

[155] Yun, W. and Bai, D. [1987]. Cost limit replacement policy under imperfect repair, *Reliability Engineering* **19**(1): 23–28.

[156] Zhang, D., Bailey, A. D. and Djurdjanovic, D. [2016]. Bayesian identification of hidden markov models and their use for condition-based monitoring, *IEEE Transactions on Reliability* **65**(3): 1471–1482.

[157] Zhang, M. and Revie, M. [2016]. Model selection with application to gamma process and inverse gaussian process, *European Safety and Reliability Conference (ESREL2016).*

[158] Zhang, X. and Gao, H. [2012]. Road maintenance optimization through a

discrete-time semi-markov decision process, *Reliability Engineering & System Safety* **103**: 110–119.

[159] Zhou, G. D., Yi, T. H. and Chen, B. [2016]. Innovative design of a health monitoring system and its implementation in a complicated long-span arch bridge, *Journal of Aerospace Engineering* **30**(2): B4016006–B4016006–17.

[160] Zhuravleva, O., Ivanov, A., Kurnosov, V., Kurnosov, K., Romantsevich, V. and Chernov, R. [2010]. Reliability estimation for semiconductor laser module ilpn-134, *Fizika i Tekhnika Poluprovodnikov* **44**(3): 377.

[161] Zou, J., Arinez, J., Chang, Q. and Lei, Y. [2016]. Opportunity window for energy saving and maintenance in stochastic production systems, *Journal of Manufacturing Science and Engineering* **138**(12): 121009–121009–9.

# Appendix A

# Toy-size problem: compare different modelling approaches

## A.1 PCM approach: lists of additional parameters and derived decisions

Table A.1: PCM: full list of unit states

| Unit state index | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states; see Table 4.1 for reference to mill state index) | $\{10, 10, 10\}$ | $\{9, 10, 10\}$ | $\{9, 9, 10\}$ | $\{9, 9, 9\}$ | $\{8, 10, 10\}$ |
| Unit state index | 6 | 7 | 8 | 9 | 10 |
| meaning (in terms of set of mill states) | $\{8, 9, 10\}$ | $\{8, 9, 9\}$ | $\{8, 8, 10\}$ | $\{8, 8, 9\}$ | $\{8, 8, 8\}$ |
| Unit state index | 11 | 12 | 13 | 14 | 15 |

Table A.1: PCM: full list of unit states

| | | | | | |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{7,10,10\}$ | $\{7,9,10\}$ | $\{7,9,9\}$ | $\{7,8,10\}$ | $\{7,8,9\}$ |
| Unit state index | 16 | 17 | 18 | 19 | 20 |
| meaning (in terms of set of mill states) | $\{7,8,8\}$ | $\{7,7,10\}$ | $\{7,7,9\}$ | $\{7,7,8\}$ | $\{7,7,7\}$ |
| Unit state index | 21 | 22 | 23 | 24 | 25 |
| meaning (in terms of set of mill states) | $\{6,10,10\}$ | $\{6,9,10\}$ | $\{6,9,9\}$ | $\{6,8,10\}$ | $\{6,8,9\}$ |
| Unit state index | 26 | 27 | 28 | 29 | 30 |
| meaning (in terms of set of mill states) | $\{6,8,8\}$ | $\{6,7,10\}$ | $\{6,7,9\}$ | $\{6,7,8\}$ | $\{6,7,7\}$ |
| Unit state index | 31 | 32 | 33 | 34 | 35 |
| meaning (in terms of set of mill states) | $\{6,6,10\}$ | $\{6,6,9\}$ | $\{6,6,8\}$ | $\{6,6,7\}$ | $\{6,6,6\}$ |
| Unit state index | 36 | 37 | 38 | 39 | 40 |
| meaning (in terms of set of mill states) | $\{5,10,10\}$ | $\{5,9,10\}$ | $\{5,9,9\}$ | $\{5,8,10\}$ | $\{5,8,9\}$ |
| Unit state index | 41 | 42 | 43 | 44 | 45 |
| meaning (in terms of set of mill states) | $\{5,8,8\}$ | $\{5,7,10\}$ | $\{5,7,9\}$ | $\{5,7,8\}$ | $\{5,7,7\}$ |

Table A.1: PCM: full list of unit states

| Unit state index | 46 | 47 | 48 | 49 | 50 |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{5, 6, 10\}$ | $\{5, 6, 9\}$ | $\{5, 6, 8\}$ | $\{5, 6, 7\}$ | $\{5, 6, 6\}$ |
| Unit state index | 51 | 52 | 53 | 54 | 55 |
| meaning (in terms of set of mill states) | $\{5, 5, 10\}$ | $\{5, 5, 9\}$ | $\{5, 5, 8\}$ | $\{5, 5, 7\}$ | $\{5, 5, 6\}$ |
| Unit state index | 56 | 57 | 58 | 59 | 60 |
| meaning (in terms of set of mill states) | $\{5, 5, 5\}$ | $\{4, 10, 10\}$ | $\{4, 9, 10\}$ | $\{4, 9, 9\}$ | $\{4, 8, 10\}$ |
| Unit state index | 61 | 62 | 63 | 64 | 65 |
| meaning (in terms of set of mill states) | $\{4, 8, 9\}$ | $\{4, 8, 8\}$ | $\{4, 7, 10\}$ | $\{4, 7, 9\}$ | $\{4, 7, 8\}$ |
| Unit state index | 66 | 67 | 68 | 69 | 70 |
| meaning (in terms of set of mill states) | $\{4, 7, 7\}$ | $\{4, 6, 10\}$ | $\{4, 6, 9\}$ | $\{4, 6, 8\}$ | $\{4, 6, 7\}$ |
| Unit state index | 71 | 72 | 73 | 74 | 75 |
| meaning (in terms of set of mill states) | $\{4, 6, 6\}$ | $\{4, 5, 10\}$ | $\{4, 5, 9\}$ | $\{4, 5, 8\}$ | $\{4, 5, 7\}$ |
| Unit state index | 76 | 77 | 78 | 79 | 80 |

Table A.1: PCM: full list of unit states

| | | | | | |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{4,5,6\}$ | $\{4,5,5\}$ | $\{4,4,10\}$ | $\{4,4,9\}$ | $\{4,4,8\}$ |
| Unit state index | 81 | 82 | 83 | 84 | 85 |
| meaning (in terms of set of mill states) | $\{4,4,7\}$ | $\{4,4,6\}$ | $\{4,4,5\}$ | $\{4,4,4\}$ | $\{3,10,10\}$ |
| Unit state index | 86 | 87 | 88 | 89 | 90 |
| meaning (in terms of set of mill states) | $\{3,9,10\}$ | $\{3,9,9\}$ | $\{3,8,10\}$ | $\{3,8,9\}$ | $\{3,8,8\}$ |
| Unit state index | 91 | 92 | 93 | 94 | 95 |
| meaning (in terms of set of mill states) | $\{3,7,10\}$ | $\{3,7,9\}$ | $\{3,7,8\}$ | $\{3,7,7\}$ | $\{3,6,10\}$ |
| Unit state index | 96 | 97 | 98 | 99 | 100 |
| meaning (in terms of set of mill states) | $\{3,6,9\}$ | $\{3,6,8\}$ | $\{3,6,7\}$ | $\{3,6,6\}$ | $\{3,5,10\}$ |
| Unit state index | 101 | 102 | 103 | 104 | 105 |
| meaning (in terms of set of mill states) | $\{3,5,9\}$ | $\{3,5,8\}$ | $\{3,5,7\}$ | $\{3,5,6\}$ | $\{3,5,5\}$ |
| Unit state index | 106 | 107 | 108 | 109 | 110 |
| meaning (in terms of set of mill states) | $\{3,4,10\}$ | $\{3,4,9\}$ | $\{3,4,8\}$ | $\{3,4,7\}$ | $\{3,4,6\}$ |

Table A.1: PCM: full list of unit states

| Unit state index | 111 | 112 | 113 | 114 | 115 |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{3,4,5\}$ | $\{3,4,4\}$ | $\{3,3,10\}$ | $\{3,3,9\}$ | $\{3,3,8\}$ |
| Unit state index | 116 | 117 | 118 | 119 | 120 |
| meaning (in terms of set of mill states) | $\{3,3,7\}$ | $\{3,3,6\}$ | $\{3,3,5\}$ | $\{3,3,4\}$ | $\{3,3,3\}$ |
| Unit state index | 121 | 122 | 123 | 124 | 125 |
| meaning (in terms of set of mill states) | $\{2,10,10\}$ | $\{2,9,10\}$ | $\{2,9,9\}$ | $\{2,8,10\}$ | $\{2,8,9\}$ |
| Unit state index | 126 | 127 | 128 | 129 | 130 |
| meaning (in terms of set of mill states) | $\{2,8,8\}$ | $\{2,7,10\}$ | $\{2,7,9\}$ | $\{2,7,8\}$ | $\{2,7,7\}$ |
| Unit state index | 131 | 132 | 133 | 134 | 135 |
| meaning (in terms of set of mill states) | $\{2,6,10\}$ | $\{2,6,9\}$ | $\{2,6,8\}$ | $\{2,6,7\}$ | $\{2,6,6\}$ |
| Unit state index | 136 | 137 | 138 | 139 | 140 |
| meaning (in terms of set of mill states) | $\{2,5,10\}$ | $\{2,5,9\}$ | $\{2,5,8\}$ | $\{2,5,7\}$ | $\{2,5,6\}$ |
| Unit state index | 141 | 142 | 143 | 144 | 145 |

Table A.1: PCM: full list of unit states

| | | | | | |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{2, 5, 5\}$ | $\{2, 4, 10\}$ | $\{2, 4, 9\}$ | $\{2, 4, 8\}$ | $\{2, 4, 7\}$ |
| Unit state index | 146 | 147 | 148 | 149 | 150 |
| meaning (in terms of set of mill states) | $\{2, 4, 6\}$ | $\{2, 4, 5\}$ | $\{2, 4, 4\}$ | $\{2, 3, 10\}$ | $\{2, 3, 9\}$ |
| Unit state index | 151 | 152 | 153 | 154 | 155 |
| meaning (in terms of set of mill states) | $\{2, 3, 8\}$ | $\{2, 3, 7\}$ | $\{2, 3, 6\}$ | $\{2, 3, 5\}$ | $\{2, 3, 4\}$ |
| Unit state index | 156 | 157 | 158 | 159 | 160 |
| meaning (in terms of set of mill states) | $\{2, 3, 3\}$ | $\{2, 2, 10\}$ | $\{2, 2, 9\}$ | $\{2, 2, 8\}$ | $\{2, 2, 7\}$ |
| Unit state index | 161 | 162 | 163 | 164 | 165 |
| meaning (in terms of set of mill states) | $\{2, 2, 6\}$ | $\{2, 2, 5\}$ | $\{2, 2, 4\}$ | $\{2, 2, 3\}$ | $\{2, 2, 2\}$ |
| Unit state index | 166 | 167 | 168 | 169 | 170 |
| meaning (in terms of set of mill states) | $\{1, 10, 10\}$ | $\{1, 9, 10\}$ | $\{1, 9, 9\}$ | $\{1, 8, 10\}$ | $\{1, 8, 9\}$ |
| Unit state index | 171 | 172 | 173 | 174 | 175 |
| meaning (in terms of set of mill states) | $\{1, 8, 8\}$ | $\{1, 7, 10\}$ | $\{1, 7, 9\}$ | $\{1, 7, 8\}$ | $\{1, 7, 7\}$ |

Table A.1: PCM: full list of unit states

| Unit state index | 176 | 177 | 178 | 179 | 180 |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{1, 6, 10\}$ | $\{1, 6, 9\}$ | $\{1, 6, 8\}$ | $\{1, 6, 7\}$ | $\{1, 6, 6\}$ |
| Unit state index | 181 | 182 | 183 | 184 | 185 |
| meaning (in terms of set of mill states) | $\{1, 5, 10\}$ | $\{1, 5, 9\}$ | $\{1, 5, 8\}$ | $\{1, 5, 7\}$ | $\{1, 5, 6\}$ |
| Unit state index | 186 | 187 | 188 | 189 | 190 |
| meaning (in terms of set of mill states) | $\{1, 5, 5\}$ | $\{1, 4, 10\}$ | $\{1, 4, 9\}$ | $\{1, 4, 8\}$ | $\{1, 4, 7\}$ |
| Unit state index | 191 | 192 | 193 | 194 | 195 |
| meaning (in terms of set of mill states) | $\{1, 4, 6\}$ | $\{1, 4, 5\}$ | $\{1, 4, 4\}$ | $\{1, 3, 10\}$ | $\{1, 3, 9\}$ |
| Unit state index | 196 | 197 | 198 | 199 | 200 |
| meaning (in terms of set of mill states) | $\{1, 3, 8\}$ | $\{1, 3, 7\}$ | $\{1, 3, 6\}$ | $\{1, 3, 5\}$ | $\{1, 3, 4\}$ |
| Unit state index | 201 | 202 | 203 | 204 | 205 |
| meaning (in terms of set of mill states) | $\{1, 3, 3\}$ | $\{1, 2, 10\}$ | $\{1, 2, 9\}$ | $\{1, 2, 8\}$ | $\{1, 2, 7\}$ |
| Unit state index | 206 | 207 | 208 | 209 | 210 |

Table A.1: PCM: full list of unit states

| | | | | | |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states) | $\{1,2,6\}$ | $\{1,2,5\}$ | $\{1,2,4\}$ | $\{1,2,3\}$ | $\{1,2,2\}$ |
| Unit state index | 211 | 212 | 213 | 214 | 215 |
| meaning (in terms of set of mill states) | $\{1,1,10\}$ | $\{1,1,9\}$ | $\{1,1,8\}$ | $\{1,1,7\}$ | $\{1,1,6\}$ |
| Unit state index | 216 | 217 | 218 | 219 | 220 |
| meaning (in terms of set of mill states) | $\{1,1,5\}$ | $\{1,1,4$ | $\{1,1,3\}$ | $\{1,1,2\}$ | $\{1,1,1\}$ |

Table A.2: PCM: derived decisions and expected value

| Initial unit state index | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Derived maintenance action | 1 | 7 | 7 | 7 | 6 |
| Expected value (£) | 27561445.32 | 27407704.37 | 27183135.36 | 26894373.16 | 27407704.37 |
| Initial unit state index | 6 | 7 | 8 | 9 | 10 |
| Derived maintenance action | 6 | 6 | 6 | 6 | 6 |
| Expected value (£) | 27183135.36 | 26894373.16 | 26962255.19 | 26662894.19 | 26231981.27 |
| Initial unit state index | 11 | 12 | 13 | 14 | 15 |
| Derived maintenance action | 14 | 14 | 14 | 6 | 6 |
| Expected value (£) | 27202144.37 | 26977575.36 | 26688813.16 | 26889750.31 | 26660153.93 |
| Initial unit state index | 16 | 17 | 18 | 19 | 20 |
| Derived maintenance action | 6 | 14 | 14 | 6 | 14 |
| Expected value (£) | 26431522.3 | 26684190.31 | 26454593.93 | 26253006.42 | 26047446.42 |
| Initial unit state index | 21 | 22 | 23 | 24 | 25 |
| Derived maintenance action | 13 | 13 | 13 | 13 | 13 |
| Expected value (£) | 27202144.37 | 26977575.36 | 26688813.16 | 26756695.19 | 26457334.19 |

Table A.2: PCM: derived decisions and expected value

| | | | | | |
|---|---|---|---|---|---|
| Initial unit state index | 26 | 27 | 28 | 29 | 30 |
| Derived maintenance action | 6 | 13 | 13 | 13 | 13 |
| Expected value (£) | 26197930.19 | 26684190.31 | 26454593.93 | 26225962.3 | 26047446.42 |
| Initial unit state index | 31 | 32 | 33 | 34 | 35 |
| Derived maintenance action | 13 | 13 | 13 | 13 | 13 |
| Expected value (£) | 26511090.14 | 26230355.66 | 25992370.19 | 25855409.74 | 25620634.97 |
| Initial unit state index | 36 | 37 | 38 | 39 | 40 |
| Derived maintenance action | 12 | 12 | 12 | 12 | 12 |
| Expected value (£) | 27202144.37 | 26977575.36 | 26688813.16 | 26756695.19 | 26457334.19 |
| Initial unit state index | 41 | 42 | 43 | 44 | 45 |
| Derived maintenance action | 12 | 12 | 12 | 12 | 12 |
| Expected value (£) | 26026421.27 | 26684190.31 | 26454593.93 | 26225962.3 | 26047446.42 |
| Initial unit state index | 46 | 47 | 48 | 49 | 50 |
| Derived maintenance action | 12 | 12 | 12 | 12 | 12 |
| Expected value (£) | 26511090.14 | 26230355.66 | 25992370.19 | 25855409.74 | 25620634.97 |
| Initial unit state index | 51 | 52 | 53 | 54 | 55 |
| Derived maintenance action | 12 | 12 | 12 | 12 | 12 |
| Expected value (£) | 26279870.47 | 25982320.01 | 25572592.17 | 25636599.77 | 25392276.77 |
| Initial unit state index | 56 | 57 | 58 | 59 | 60 |
| Derived maintenance action | 12 | 11 | 11 | 11 | 11 |
| Expected value (£) | 24993210.54 | 27202144.37 | 26977575.36 | 26688813.16 | 26756695.19 |
| Initial unit state index | 61 | 62 | 63 | 64 | 65 |
| Derived maintenance action | 11 | 11 | 11 | 11 | 11 |
| Expected value (£) | 26457334.19 | 26026421.27 | 26684190.31 | 26454593.93 | 26225962.3 |
| Initial unit state index | 66 | 67 | 68 | 69 | 70 |
| Derived maintenance action | 11 | 11 | 11 | 11 | 11 |
| Expected value (£) | 26047446.42 | 26511090.14 | 26230355.66 | 25992370.19 | 25855409.74 |
| Initial unit state index | 71 | 72 | 73 | 74 | 75 |
| Derived maintenance action | 11 | 11 | 11 | 11 | 11 |
| Expected value (£) | 25620634.97 | 26279870.47 | 25982320.01 | 25572592.17 | 25636599.77 |

Table A.2: PCM: derived decisions and expected value

| | | | | | |
|---|---|---|---|---|---|
| Initial unit state index | 76 | 77 | 78 | 79 | 80 |
| Derived maintenance action | 11 | 11 | 11 | 11 | 11 |
| Expected value (£) | 25392276.77 | 24993210.54 | 25350390.78 | 25088665.22 | 24856798.59 |
| Initial unit state index | 81 | 82 | 83 | 84 | 85 |
| Derived maintenance action | 11 | 11 | 11 | 11 | 10 |
| Expected value (£) | 24577919.96 | 24365010.68 | 24146824.29 | 22665926.83 | 27202144.37 |
| Initial unit state index | 86 | 87 | 88 | 89 | 90 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 26977575.36 | 26688813.16 | 26756695.19 | 26457334.19 | 26026421.27 |
| Initial unit state index | 91 | 92 | 93 | 94 | 95 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 26684190.31 | 26454593.93 | 26225962.3 | 26047446.42 | 26511090.14 |
| Initial unit state index | 96 | 97 | 98 | 99 | 100 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 26230355.66 | 25992370.19 | 25855409.74 | 25620634.97 | 26279870.47 |
| Initial unit state index | 101 | 102 | 103 | 104 | 105 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 25982320.01 | 25572592.17 | 25636599.77 | 25392276.77 | 24993210.54 |
| Initial unit state index | 106 | 107 | 108 | 109 | 110 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 25350390.78 | 25088665.22 | 24856798.59 | 24577919.96 | 24365010.68 |
| Initial unit state index | 111 | 112 | 113 | 114 | 115 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 24146824.29 | 22665926.83 | 25240844.87 | 24953058.36 | 24649657.23 |
| Initial unit state index | 116 | 117 | 118 | 119 | 120 |
| Derived maintenance action | 10 | 10 | 10 | 10 | 10 |
| Expected value (£) | 24466892.94 | 24234112.35 | 23943765.08 | 22555812.05 | 22399097.27 |
| Initial unit state index | 121 | 122 | 123 | 124 | 125 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 27202144.37 | 26977575.36 | 26688813.16 | 26756695.19 | 26457334.19 |

Table A.2: PCM: derived decisions and expected value

| | | | | | |
|---|---|---|---|---|---|
| Initial unit state index | 126 | 127 | 128 | 129 | 130 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 26026421.27 | 26684190.31 | 26454593.93 | 26225962.3 | 26047446.42 |
| Initial unit state index | 131 | 132 | 133 | 134 | 135 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 26511090.14 | 26230355.66 | 25992370.19 | 25855409.74 | 25620634.97 |
| Initial unit state index | 136 | 137 | 138 | 139 | 140 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 26279870.47 | 25982320.01 | 25572592.17 | 25636599.77 | 25392276.77 |
| Initial unit state index | 141 | 142 | 143 | 144 | 145 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 24993210.54 | 25350390.78 | 25088665.22 | 24856798.59 | 24577919.96 |
| Initial unit state index | 146 | 147 | 148 | 149 | 150 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 24365010.68 | 24146824.29 | 22665926.83 | 25240844.87 | 24953058.36 |
| Initial unit state index | 151 | 152 | 153 | 154 | 155 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 24649657.23 | 24466892.94 | 24234112.35 | 23943765.08 | 22555812.05 |
| Initial unit state index | 156 | 157 | 158 | 159 | 160 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 22399097.27 | 25041441.05 | 24747763.12 | 24385229.7 | 24270723.21 |
| Initial unit state index | 161 | 162 | 163 | 164 | 165 |
| Derived maintenance action | 9 | 9 | 9 | 9 | 9 |
| Expected value (£) | 24034312.72 | 23684271.57 | 22374400.98 | 22182608.97 | 21936494.82 |
| Initial unit state index | 166 | 167 | 168 | 169 | 170 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 27202144.37 | 26977575.36 | 26688813.16 | 26756695.19 | 26457334.19 |
| Initial unit state index | 171 | 172 | 173 | 174 | 175 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 26026421.27 | 26684190.31 | 26454593.93 | 26225962.3 | 26047446.42 |

Table A.2: PCM: derived decisions and expected value

| | | | | | |
|---|---|---|---|---|---|
| Initial unit state index | 176 | 177 | 178 | 179 | 180 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 26511090.14 | 26230355.66 | 25992370.19 | 25855409.74 | 25620634.97 |
| Initial unit state index | 181 | 182 | 183 | 184 | 185 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 26279870.47 | 25982320.01 | 25572592.17 | 25636599.77 | 25392276.77 |
| Initial unit state index | 186 | 187 | 188 | 189 | 190 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 24993210.54 | 25350390.78 | 25088665.22 | 24856798.59 | 24577919.96 |
| Initial unit state index | 191 | 192 | 193 | 194 | 195 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 24365010.68 | 24146824.29 | 22665926.83 | 25240844.87 | 24953058.36 |
| Initial unit state index | 196 | 197 | 198 | 199 | 200 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 24649657.23 | 24466892.94 | 24234112.35 | 23943765.08 | 22555812.05 |
| Initial unit state index | 201 | 202 | 203 | 204 | 205 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 22399097.27 | 25041441.05 | 24747763.12 | 24385229.7 | 24270723.21 |
| Initial unit state index | 206 | 207 | 208 | 209 | 210 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 24034312.72 | 23684271.57 | 22374400.98 | 22182608.97 | 21936494.82 |
| Initial unit state index | 211 | 212 | 213 | 214 | 215 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 21740228.19 | 21400787.94 | 21185664.05 | 20761654.03 | 20484793.1 |
| Initial unit state index | 216 | 217 | 218 | 219 | 220 |
| Derived maintenance action | 8 | 8 | 8 | 8 | 8 |
| Expected value (£) | 20287501.06 | 18291897.07 | 18167769.68 | 17997225.49 | 12980370.85 |

## A.2 Myopic approach: derived decisions comparison

Table A.3: Myopic approach: derived decisions comparison

| Initial unit state index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 1 | 7 | 7 | 7 | 6 | 6 | 6 | 6 | 6 | 6 |
| Myopic maintenance choice | 1 | 1 | 1 | 1 | 1 | 1 | 6 | 6 | 6 | 6 |
| Loss of expected value (%) | 0 | 0.03 | 0.12 | 0.26 | 0.58 | 0.87 | 0 | 0 | 0 | 0 |
| Initial unit state index | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Optimal maintenance choice | 14 | 14 | 14 | 6 | 6 | 6 | 14 | 14 | 6 | 14 |
| Myopic maintenance choice | 1 | 1 | 1 | 1 | 6 | 6 | 1 | 1 | 6 | 1 |
| Loss of expected value (%) | 0.59 | 0.39 | 0.28 | 0.65 | 0 | 0 | 1.05 | 0.85 | 0 | 1.24 |
| Initial unit state index | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 6 | 13 | 13 | 13 | 13 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Myopic maintenance choice | 1 | 1 | 1 | 1 | 6 | 6 | 1 | 1 | 6 | 1 |
| Loss of expected value (%) | 1.00 | 1.06 | 1.20 | 1.03 | 0.08 | 0 | 1.53 | 1.61 | 0.63 | 1.75 |
| Initial unit state index | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 12 | 12 | 12 | 12 | 12 |
| Myopic maintenance choice | 1 | 1 | 6 | 1 | 1 | 1 | 1 | 4 | 6 | 6 |
| Loss of expected value (%) | 1.62 | 1.74 | 0.64 | 1.80 | 1.90 | 1.59 | 1.88 | 1.11 | 1.01 | 1.02 |
| Initial unit state index | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| Myopic maintenance choice | 6 | 1 | 4 | 6 | 4 | 1 | 4 | 6 | 4 | 4 |
| Loss of expected value (%) | 0.95 | 2.09 | 1.54 | 1.46 | 1.72 | 2.38 | 1.61 | 1.52 | 1.72 | 1.75 |
| Initial unit state index | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 11 | 11 | 11 | 11 |
| Myopic maintenance choice | 4 | 4 | 6 | 4 | 4 | 4 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 1.53 | 1.56 | 1.46 | 1.73 | 1.76 | 1.81 | 5.51 | 5.46 | 5.57 | 5.28 |
| Initial unit state index | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Myopic maintenance choice | 6 | 6 | 1 | 1 | 6 | 1 | 1 | 1 | 6 | 1 |
| Loss of expected value (%) | 4.40 | 3.70 | 6.57 | 6.54 | 5.50 | 7.48 | 6.53 | 6.62 | 5.47 | 7.40 |
| Initial unit state index | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Myopic maintenance choice | 1 | 1 | 4 | 6 | 4 | 4 | 4 | 1 | 1 | 6 |
| Loss of expected value (%) | 7.44 | 6.27 | 5.51 | 4.77 | 6.45 | 6.38 | 5.75 | 9.21 | 9.22 | 7.99 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Initial unit state index | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 10 | 10 |
| Myopic maintenance choice | 1 | 1 | 4 | 1 | 1 | 1 | 1 | 6 | 6 | 6 |
| Loss of expected value (%) | 10.17 | 10.13 | 9.01 | 12.34 | 5.67 | 5.81 | 6.10 | 4.90 | 4.91 | 4.50 |
| Initial unit state index | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Myopic maintenance choice | 1 | 1 | 6 | 1 | 1 | 1 | 6 | 1 | 1 | 4 |
| Loss of expected value (%) | 6.75 | 6.91 | 5.92 | 7.67 | 6.86 | 7.14 | 5.97 | 7.75 | 7.95 | 5.99 |
| Initial unit state index | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Myopic maintenance choice | 4 | 6 | 4 | 4 | 4 | 1 | 1 | 6 | 1 | 1 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Loss of expected value (%) | 6.03 | 5.57 | 6.87 | 6.90 | 6.56 | 9.39 | 9.65 | 8.43 | 10.38 | 10.54 |
| Initial unit state index | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Myopic maintenance choice | 4 | 1 | 1 | 1 | 6 | 1 | 1 | 4 | 1 | 1 |
| Loss of expected value (%) | 9.47 | 12.63 | 9.45 | 9.82 | 8.30 | 10.46 | 10.73 | 9.36 | 12.82 | 13.08 |
| Initial unit state index | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | 129 | 130 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Myopic maintenance choice | 1 | 1 | 2 | 6 | 6 | 6 | 1 | 1 | 6 | 1 |
| Loss of expected value (%) | 6.15 | 6.47 | 6.05 | 5.64 | 5.68 | 5.52 | 7.22 | 7.59 | 6.67 | 8.14 |
| Initial unit state index | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Myopic maintenance choice | 1 | 2 | 6 | 1 | 2 | 4 | 4 | 6 | 4 | 4 |
| Loss of expected value (%) | 7.52 | 7.10 | 6.74 | 8.43 | 7.93 | 6.72 | 6.81 | 6.58 | 7.61 | 7.68 |
| Initial unit state index | 141 | 142 | 143 | 144 | 145 | 146 | 147 | 148 | 149 | 150 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Myopic maintenance choice | 4 | 1 | 2 | 6 | 1 | 2 | 4 | 1 | 1 | 2 |
| Loss of expected value (%) | 7.59 | 9.83 | 9.51 | 9.16 | 10.84 | 10.43 | 10.21 | 13.13 | 10.21 | 9.58 |
| Initial unit state index | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Myopic maintenance choice | 6 | 2 | 2 | 4 | 2 | 2 | 1 | 2 | 6 | 2 |
| Loss of expected value (%) | 9.17 | 10.35 | 10.50 | 10.25 | 12.47 | 12.56 | 10.48 | 9.63 | 9.20 | 10.35 |
| Initial unit state index | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 2 | 4 | 2 | 2 | 2 | 1 | 1 | 1 | 6 | 6 |
| Loss of expected value (%) | 10.55 | 10.30 | 12.53 | 12.66 | 12.76 | 18.30 | 18.89 | 19.50 | 17.98 | 18.34 |
| Initial unit state index | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 | 179 | 180 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 6 | 1 | 1 | 6 | 1 | 1 | 1 | 6 | 1 | 1 |
| Loss of expected value (%) | 17.81 | 20.38 | 21.00 | 20.05 | 22.34 | 20.92 | 21.51 | 20.40 | 22.85 | 23.35 |
| Initial unit state index | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 4 | 4 | 6 | 4 | 4 | 4 | 1 | 1 | 6 | 1 |
| Loss of expected value (%) | 20.06 | 20.45 | 19.86 | 22.01 | 22.35 | 21.90 | 25.94 | 26.54 | 25.58 | 27.96 |

Table A.3: Myopic approach: derived decisions comparison

| Initial unit state index | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 1 | 4 | 1 | 1 | 1 | 6 | 1 | 1 | 4 | 1 |
| Loss of expected value (%) | 28.46 | 27.65 | 33.21 | 26.12 | 26.77 | 25.46 | 28.16 | 28.69 | 27.55 | 33.45 |
| Initial unit state index | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 1 | 1 | 2 | 6 | 2 | 2 | 4 | 2 | 2 | 2 |
| Loss of expected value (%) | 33.72 | 26.21 | 26.10 | 25.35 | 27.65 | 28.08 | 27.47 | 32.98 | 32.96 | 32.98 |
| Initial unit state index | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Myopic maintenance choice | 1 | 1 | 6 | 8 | 8 | 4 | 8 | 8 | 8 | 8 |

Table A.3: Myopic approach: derived decisions comparison

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Loss of expected value (%) | 38.07 | 38.67 | 37.76 | 0 | 0 | 39.96 | 0 | 0 | 0 | 0 |

## A.3 Perfect correlation approach: lists of additional parameters and derived decisions comparison

| Unit state index | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| meaning (in terms of set of mill states; see Table 4.4 for reference to mill state index) | $\{4,4,4\}$ | $\{3,4,4\}$ | $\{3,3,4\}$ | $\{3,3,3\}$ | $\{2,4,4\}$ |
| Unit state index | 6 | 7 | 8 | 9 | 10 |
| meaning (in terms of set of mill states) | $\{2,3,4\}$ | $\{2,3,3\}$ | $\{2,2,4\}$ | $\{2,2,3\}$ | $\{2,2,2\}$ |
| Unit state index | 11 | 12 | 13 | 14 | 15 |
| meaning (in terms of set of mill states) | $\{1,4,4\}$ | $\{1,3,4\}$ | $\{1,3,3\}$ | $\{1,2,4\}$ | $\{1,2,3\}$ |
| Unit state index | 16 | 17 | 18 | 19 | 20 |
| meaning (in terms of set of mill states) | $\{1,2,2\}$ | $\{1,1,4\}$ | $\{1,1,3\}$ | $\{1,1,2\}$ | $\{1,1,1\}$ |

Table A.4: PM: full list of unit states

| Unit state index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Derived maintenance action | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 |
| Unit state index | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Derived maintenance action | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

Table A.5: Perfect correlation approach: derived decisions

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Initial unit state index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Optimal maintenance choice | 1 | 7 | 7 | 7 | 6 | 6 | 6 | 6 | 6 | 6 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 0 | 0.03 | 0.12 | 0.26 | 0.58 | 0.87 | 1.15 | 1.66 | 1.98 | 2.78 |
| Initial unit state index | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Optimal maintenance choice | 14 | 14 | 14 | 6 | 6 | 6 | 14 | 14 | 6 | 14 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 0.59 | 0.39 | 0.28 | 0.65 | 0.97 | 1.80 | 1.05 | 0.85 | 0.72 | 1.24 |
| Initial unit state index | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 6 | 13 | 13 | 13 | 13 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 1.00 | 1.06 | 1.20 | 1.03 | 1.26 | 2.05 | 1.53 | 1.61 | 1.60 | 1.75 |
| Initial unit state index | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 12 | 12 | 12 | 12 | 12 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 1.62 | 1.74 | 1.82 | 1.80 | 1.90 | 1.59 | 1.88 | 2.19 | 2.61 | 2.93 |
| Initial unit state index | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 3.67 | 2.09 | 2.42 | 3.18 | 2.35 | 2.38 | 2.70 | 3.48 | 2.61 | 2.85 |
| Initial unit state index | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 11 | 11 | 11 | 11 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 11 | 11 | 11 | 11 |
| Loss of expected value (%) | 3.05 | 3.41 | 4.12 | 3.39 | 3.66 | 4.41 | 0 | 0 | 0 | 0 |
| Initial unit state index | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Maintenance choice (perfect correlation approach) | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 11 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | 129 | 130 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 141 | 142 | 143 | 144 | 145 | 146 | 147 | 148 | 149 | 150 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 | 179 | 180 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |

Table A.6: Perfect correlation approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## A.4 Deterministic wearing-off approach: derived decisions comparison

| Unit state index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Derived maintenance action | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 |
| Unit state index | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Derived maintenance action | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

Table A.7: Deterministic wearing-off approach: derived decisions

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| Initial unit state index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 1 | 7 | 7 | 7 | 6 | 6 | 6 | 6 | 6 | 6 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Loss of expected value (%) | 0 | 0.03 | 0.12 | 0.26 | 0.58 | 0.87 | 1.15 | 1.66 | 1.98 | 2.78 |
| Initial unit state index | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Optimal maintenance choice | 14 | 14 | 14 | 6 | 6 | 6 | 14 | 14 | 6 | 14 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 0.59 | 0.39 | 0.28 | 0.65 | 0.97 | 1.80 | 1.05 | 0.85 | 0.72 | 1.24 |
| Initial unit state index | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 6 | 13 | 13 | 13 | 13 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 1.00 | 1.06 | 1.20 | 1.03 | 1.26 | 2.05 | 1.53 | 1.61 | 1.60 | 1.75 |
| Initial unit state index | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| Optimal maintenance choice | 13 | 13 | 13 | 13 | 13 | 12 | 12 | 12 | 12 | 12 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 1.62 | 1.74 | 1.82 | 1.80 | 1.90 | 1.59 | 1.88 | 2.19 | 2.61 | 2.93 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| Initial unit state index | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 3.67 | 2.09 | 2.42 | 3.18 | 2.35 | 2.38 | 2.70 | 3.48 | 2.61 | 2.85 |
| Initial unit state index | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| Optimal maintenance choice | 12 | 12 | 12 | 12 | 12 | 12 | 11 | 11 | 11 | 11 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 3.05 | 3.41 | 4.12 | 3.39 | 3.66 | 4.41 | 5.51 | 5.46 | 5.57 | 5.28 |
| Initial unit state index | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 5.56 | 5.88 | 6.57 | 6.54 | 6.38 | 7.48 | 6.53 | 6.62 | 6.63 | 7.40 |
| Initial unit state index | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 11 | 11 | 11 |
| Loss of expected value (%) | 7.44 | 6.27 | 6.57 | 6.86 | 7.22 | 7.44 | 7.75 | 0 | 0 | 0 |
| Initial unit state index | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 |
| Optimal maintenance choice | 11 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 11 | 11 | 11 | 11 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 5.67 | 5.81 | 6.10 | 6.06 | 6.39 | 6.91 |
| Initial unit state index | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 6.75 | 6.91 | 7.18 | 7.67 | 6.86 | 7.14 | 7.47 | 7.75 | 7.95 | 7.05 |
| Initial unit state index | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 7.42 | 7.89 | 8.02 | 8.30 | 8.81 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 |
| Optimal maintenance choice | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Maintenance choice (perfect correlation approach) | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | 129 | 130 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 6.15 | 6.47 | 6.91 | 7.05 | 7.40 | 8.03 | 7.22 | 7.59 | 8.18 | 8.14 |
| Initial unit state index | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Loss of expected value (%) | 7.52 | 7.95 | 8.48 | 8.43 | 8.75 | 8.03 | 8.43 | 9.02 | 9.03 | 9.33 |
| Initial unit state index | 141 | 142 | 143 | 144 | 145 | 146 | 147 | 148 | 149 | 150 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Maintenance choice (perfect correlation approach) | 1 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 9.94 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 |
| Optimal maintenance choice | 9 | 9 | 9 | 9 | 9 | 8 | 8 | 8 | 8 | 8 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 9 | 9 | 9 | 9 | 9 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 | 179 | 180 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |

Table A.8: Deterministic wearing-off approach: derived decisions comparison

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Initial unit state index | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 |
| Optimal maintenance choice | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Maintenance choice (perfect correlation approach) | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Loss of expected value (%) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |