Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

Trym Haakon Andenes Blaauw
Jakob Erikstad

Power Grid Inspection and Maintenance Optimization

A Stochastic Dynamic Programming Approach

Master's thesis in Industrial Economics and Technology Management
Supervisor: Peter Schütz
February 2021

Photo: Jason Blackeye

NTNU
Norwegian University of
Science and Technology

Trym Haakon Andenes Blaauw
Jakob Erikstad

Power Grid Inspection and Maintenance Optimization

A Stochastic Dynamic Programming Approach

Master's thesis in Industrial Economics and Technology Management
Supervisor: Peter Schütz
February 2021

Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

NTNU
Norwegian University of
Science and Technology

Preface

This Master's thesis concludes our Master of Science at the Norwegian University of Science and Technology (NTNU). The specialization is Managerial Economics and Operations Research at the Department of Industrial Economics and Technology Management. This thesis is motivated by the operations of Wiseline AS, and written in collaboration with them. It continues the work of our project report from the fall semester of 2019 (Blaauw and Erikstad, 2019) and seeks to optimize both inspection and maintenance decisions for a power grid operator.

We would like to thank our supervisor Peter Schütz for providing valuable guidance, feedback and discussions throughout our work with this thesis. We truly appreciate Wiseline's willingness to aid with data, and insight in the field of power grid operation through discussions.

Trondheim, February 1, 2020

Trym Haakon Andenes Blaauw | Jakob Erikstad

Abstract

This Master's thesis studies operations research in the context of maintenance and inspection on the power grid line. The potential of lowering operating costs by optimizing inspection and maintenance decisions motivates this study. The problem is to plan when to inspect a utility mast and which maintenance to perform based on the information inspections reveal. A utility mast is a multi-unit system composed of several components. A variety of available decisions and random events affects a mast's condition over time. Today, power grid operators do not utilize comprehensive data-analysis when making inspection and maintenance decisions. This indicates that mathematical models should be explored, aiming to reduce costs from operating the power grid.

We propose two models for solving the problem. One is restricted to periodic inspection intervals of fixed length. The other model allows sequential inspection decisions. That is, deciding when to inspect next at each inspection. Both models are solved to optimality using stochastic dynamic programming. They return optimal policies, which we study on a four-component utility mast. To the best of our knowledge, the literature does not consider optimal sequential inspection and maintenance optimization for multi-unit systems.

To enable decision-support for larger systems, we propose a heuristic that uses our sequential inspection model to combine solutions for smaller systems and derives an optimal policy for the four-component mast. The heuristic performs almost as well as our two models and is a good starting point for future research looking to apply maintenance optimization to real-life cases.

The findings we discuss in this thesis show a significant potential to reduce the power grid operator's total costs by applying inspection and maintenance optimization. Our proposed models provide optimal inspection and maintenance decisions for a sub-set of critical mast components, and our proposed heuristic lessens the gap between theory and real-life usage.

Sammendrag

Denne masteroppgaven studerer operasjonsanalyse i kontekst av inspeksjon og vedlikehold av strømnettet. Den motiveres av potensialet for besparelser ved bruk av optimale inspeksjons- og vedlikeholdsbeslutninger. Problemet er å planlegge når strømmaster skal inspiseres, og hvilket vedlikehold som skal gjennomføres basert på informasjon fra inspeksjonene. En strømmast beskrives som et system sammensatt av flere komponenter, der ulike beslutninger og tilfeldige hendelser påvirker komponentenes tilstandsutvikling over tid. I dag benyttes omfattende dataanalyse lite av nettselskapene til å støtte inspeksjons- og vedlikeholdsbeslutninger. Dette indikerer at optimeringsmodeller bør utforskes, med formål om å redusere kostnader tilknyttet drift av strømnettet.

Vi presenterer to modeller for å løse inspeksjons- og vedlikeholdsproblemet. En har faste inspeksjonsintervaller av en gitt lengde, mens den andre tillater at inspeksjonsintervallene settes sekvensielt, altså løpende gjennom en beslutningsperiode. Begge modellene er løst til optimalitet ved bruk av stokastisk dynamisk programmering. De returnerer optimale inspeksjons- og vedlikeholdsregimer, og for å utforske regimene ser vi på et mast-system bestående av fire komponenter. Så vidt vi vet beskriver ingen litteratur optimale løsningsmetoder for sekvensielle inspeksjons- og vedlikeholds beslutninger på flerkomponent-problemer.

For å muliggjøre beslutningsstøtte for større systemer lager vi en heuristikk som bruker vår sekvensielle modell til å løse en rekke mindre problemer, og kombinerer disse til en løsning for mast-systemet med fire komponenter. Denne heuristikkens regimer presterer nesten like godt som de optimale regimene fra våre modeller, og er et godt utgangspunkt for å videre utforske hvordan stokastisk dynamiske programmer kan brukes på større og mer virkelighetsnære problemer.

Våre funn indikerer et potensial for at nettselskapene ved bruk av optimering kan redusere sine kostnader tilknyttet drift av strømnettet. Våre modeller gir optimale inspeksjons- og vedlikeholdsbeslutninger for mindre system bestående av kritiske mastkomponenter, og vår foreslåtte heuristikk reduserer avstanden mellom teori og praksis.

Contents

Figures

Tables

Chapter 1

Introduction

In this thesis, we study the Utility Mast Inspection and Maintenance Problem on the Norwegian power grid. The problem involves deciding when to inspect a mast and which parts of the mast to maintain based on their condition, to minimize the costs of operating the grid. These costs include the cost of inspection and maintenance, as well as the costs associated with mast failures, such as power interruption and unplanned repairs.

A mast's condition worsens over time in a stochastic manner, and travelling out to maintain a mast has a fixed cost independent of which parts of the mast one is travelling out to fix. This uncertainty and the economic dependency between components complicate the problem. When inspecting a mast, it may be optimal to maintain any of the components only if one maintains several components at once.

A power grid ensures electricity supply to individual consumers and essential societal functions such as hospitals, research facilities and educational institutions. Managing the power grid includes monitoring the grid's condition and renewing or maintaining it when needed. In 2019, a total of 115 grid companies operated the Norwegian power grid. The transmission grid had a length of over 351 000 kilometres with a book value of 132 billion NOK, and the local distribution grid had 3.2 million customers. In the same year, cost of maintenance and operation of the grid totalled to 9.6 billion NOK. The costs from interruptions in electricity delivery were 711 million NOK (NVE, 2020b). Operating and maintaining the power grid had an average cost of 2 343 NOK per customer (NVE, 2020a).

In Norway, the tariffs charged to consumers are set based on the grid operators' collective costs of operating the power grid. Thus, cost reduction of grid line management yields a direct societal interest in grid operation efficiency. Furthermore, reliable delivery of power is important for both the operators and the consumers. In addition to increased consumer satisfaction, fewer and shorter interruptions also mean reduced penalty costs that consumers take part in through the rental of local distribution grids.

Grid management also has environmental implications. Overly conservative strategies regarding maintenance and inspection will mean unnecessary travelling across the whole grid and overconsumption of mast parts. Efficient decisions regarding the replacement or repair of mast parts may increase their lifetime. Consequently, such decisions reduce waste. Several industry experts claim that the power grid reliability will not be significantly affected by some strategies that are less conservative (Bakken, 2019). These claims imply a potential for better inspection and maintenance decisions that will have a positive socioeconomic and environmental impact.

The importance of modelling the maintenance of deteriorating systems was first acknowledged in the 1940s due to industrial and medical applications (Thomas et al., 1991). The study of optimizing decisions related to inspection and maintenance through mathematical programming begun in the 1960s. Derman (1963) published an article that model the replacement of a single component after an inspection reveals its condition. Today, publications typically concern business-specific modelling of maintenance and inspection optimization problems.

As for power grid management, there are few publications concerning maintenance optimization. To the best of the authors' knowledge, the few existing publications within the field of maintenance optimization do not consider optimal inspection and maintenance of utility masts, but rather optimal grid investments (Lim and Han, 2018) or balancing supply and demand through allocation and maintenance of generators (Xiao and Cao, 2020).

As technical systems have evolved, and we increasingly rely on different equipment, the importance of effective maintenance activities is growing (de Jonge and Scarf, 2020). Today, society depends on the power delivery system, a complex and critical infrastructure (Kiel and G. H. Kjølle, 2019). The Norwegian power grid industry standard is thorough inspection every tenth year and aerial observation every year. Bakken (2019) claims that the same risk level can be achieved by aerial inspection every *other* year which would save the operators a total of 200 million NOK per year.

The research literature on maintenance and inspection optimization investigates both single- and multi-unit systems with various possible decisions and uncertainty factors. Derman (1963) studied the problem of replacing a single component following an inspection. However, as a consequence of better techniques for analyzing complex systems, multi-unit systems has become more relevant. Lugtigheid et al. (2008) consider a setup cost, shared between all components maintained at the same time. Dynamic programming has been applied to both single-unit and multi-unit systems (see Chu et al. (1998) and Korpijärvi and Kortelainen (2009)). Several models are developed for optimizing both inspection and maintenance decisions. Typically these models consider a periodic inspection interval, especially for multi-unit systems such as the one from Babishin and Taghipour (2016). Some models allow for a more flexible sequential inspection schedule. H. Ellis et al. (1995) optimizes such an inspection schedule for a bridge. However, optimal sequential inspection schedules are suggested for future research, especially for multi-unit systems.

We model the multi-unit Utility Mast Inspection and Maintenance Problem (UMIMP) in two ways that differ with respect to inspection decisions' flexibility. The problem considers inspections and maintenances over a planning horizon. One model requires an equal period of time to pass between all inspections, while the other enables setting the next inspection when conducting an inspection. Both models are multi-stage decision processes that yield optimal inspection and maintenance decisions, adapting with events that may occur over the planning horizon. We derive optimal decisions through stochastic dynamic programming, incorporating uncertainty of future mast conditions and potential economic dependencies between components.

Furthermore, we propose a heuristic to combat the curse of dimensionality associated with stochastic dynamic programming, using one of our models to solve small problem instances, then combining the solutions to solve larger problems. We find that optimal maintenance and inspection decisions have significant potential for reducing the cost of grid line operation. Bridging the gap between theoretical models and real-life application is challenging, but should be further researched. Our heuristic highlights a possibility of approaching large problem-instances, and may be used as a starting point for power grid optimization.

The remainder of this thesis is structured as follows: Chapter 2 introduces the Norwegian power grid and its utility masts and introduce the terms "maintenance" and "inspection" in relation to maintenance optimization. Chapter 3 reviews maintenance optimization literature with a particular focus on the role of inspection and explains the theory behind Markov decision processes and dynamic programming. We provide a detailed description of the UMIMP in Chapter 4 and present the mathematical models that solve it in Chapter 5. In Chapter 6 we present the case company, Wiseline AS, and define a number of cases that are analyzed in Chapter 7. Future research topics are suggested in Chapter 8 before we present our concluding remarks in Chapter 9.

Chapter 2

Background

This chapter aims to give the reader an introduction to some of the topics addressed in this thesis. Section 2.1 presents the Norwegian power grid's characteristics and regulations, before Section 2.2 focuses on the utility mast and how its components deteriorate. Section 2.3 defines maintenance and highlights its importance, presents decisions related to maintenance and introduces maintenance optimization. Section 2.4 introduces inspection and presents its relation to maintenance optimization. The last section, Section 2.5, focuses on inspection and maintenance of the Norwegian power grid and illustrates this process.

2.1 The Norwegian power grid

This section introduces the Norwegian power grid characteristics and presents the power grid regulations imposed on the operators.

2.1.1 Norwegian power grid characteristics

In Norway, the electrical grid consists of three different levels. These are, hierarchically ordered from the top level to the bottom level, the transmission grid, the regional grid, and the distribution grid. Norway has a single, designated transmission system operator (TSO), Statnett, owning 94% of the transmission grid and renting the remaining 6%. The regional grid operators own the last 6%, operating within production and turnover. The transmission grid has a span of about 11 000 km (Ministry of Petroleum and Energy, 2019).

The regional grid is the mid-level link between the transmission and distribution grids, spanning over 19 000 km. It also supplies some high-priority end customers such as hospitals and airports (Reiten et al., 2014). In contrast to the transmission grid having only a single TSO, around 70 different actors operate the regional grid (Rosvold, 2020).

Supplying almost all small end-users such as households, in addition to commercial and industrial players, the distribution grid has a far longer span than the other levels combined of about 316 000 km. This grid level is divided into a low- and high-voltage segment at 1kV, with the high-voltage distribution grid spanning about 100 000km. The low-voltage grid is distributed to ordinary customers, normally carrying 400V or 230V. (Ministry of Petroleum and Energy, 2019).

In 2018, the cost associated with operating and maintaining the power grid was a total of 9,7 billion NOK (NVE, 2020a). This cost amounts to 19 000 NOK per kilometre of the electrical grid line, every year. The term "Operating and maintenance cost" is frequently used by The Norwegian Water Resources and Energy Directorate (NVE), including costs such as salary and staff cost, system services, cost of goods, losses on receivables, internal priced services, overhead cost and other operating costs (Syvertsen et al., 2018). Consequently, the term also includes any costs associated with the grid line inspection to reveal the needs of maintenance, and maintenance to ensure safe and reliable power delivery.

When the power grid disconnects, we have an energy delivery *interruption*. Several thousand interruptions occur on the Norwegian grid line every year. Some planned maintenances cause, while other interruptions may be unexpected and a result of utility line failure. Such interruptions mean undelivered to parts of the society, which is costly (Mjølnerød, 2019). In 2018, a total of 28 761 of interruptions occurred on the distribution grid. 10 798 of these were unplanned interruptions. The interruptions resulted in a total of 17 919 MWh of undelivered electricity (Statnett, 2019). According to statistics from Hafslund Nett from 2001-2007, their consumers will experience an average of 40 minutes interruption in their electricity supply every year. An interruption has a mean time of 50 minutes (G. H. Kjølle et al., 2012).

The replacement of a standard wooden pole in a utility mast cost about 18 000 NOK (Rasjonell Elektrisk Nettvirksomhet AS, 2019). Research suggests that the maintenance strategy of wooden poles on the Norwegian grid line is overly conservative. The frequency of wooden pole replacements could be postponed by 20 years on average. If the lifetime of today's wooden poles were extended with 20 years, the direct savings would be around 150 billion NOK (Solvang and Foros, 2019). This highlights the potential savings of more efficient maintenance on just one of the components that comprise a utility mast.

2.1.2 Regulating the Norwegian power grid operators

The Norwegian power grid regulations primarily concern the safety of the surroundings. They state that electrical facilities should not pose a threat to life, health or property (Nordnes, 2011). The regulations also require frequent *enough* inspection of the power grid line to achieve an acceptable risk of a power outage. Today, this means a minimum of yearly aerial observations and a more thorough inspection of the masts' tops in addition to checking them for rot at least every tenth year. The operators may inspect more rarely if they can show a risk analysis of their grid that proves an acceptable risk level associated with the proposed inspection plan. However, grid operators are free to inspect more often (Bakken, 2019).

As for maintaining the power grid, no regulations enforce maintenance. Instead, NVE use penalty costs that enforce sufficient grid reliability. The network operators must pay a penalty cost if they fail to supply energy to their grid line's consumers. This cost is called *Cost of Energy Not Supplied* (CENS). Penalty cost also may be enforced if safety measures are not maintained properly. Additionally, all Norwegian operators of electrical facilities are subject to rules regarding health, safety and environment (Supervisory, 2020). Regardless, there are no time-based requirements enforcing action to decrease the risk of such violations.

Most Norwegian households are connected to one grid line with a single responsible operator. As a consequence of this monopoly situation, NVE strictly regulates the grid companies and Statnett. They divide their regulations into two types: *Direct regulations* and *economic revenue regulations* (NVE, 2019).

Direct regulations define standards, roles and procedures. NVE monitor that the power grid operators comply with these regulations. NVE also hand out fines to those who violate these regulations. Maintenance is one measure that grid line operators must take to remain compliant, while inspection typically identifies potential and existing violations that need attending.

Economic revenue regulations intend to prevent the operators from exploiting the monopoly situation. The grid line operators receive an annual allowed revenue cap, including a fair tariff that the grid line consumers pay. The revenue ensures a reasonable return on investment for the power grid operators. Furthermore, the consumers' tariffs reflect the cost of operations for the power grid operators, and thus streamlining the operation will be beneficial for both operators and consumers in the long run.

The CENS costs imposed throughout a year, are deducted from the allowed revenues(Langset et al., 2001). The CENS cost is calculated based on where interruptions occur and the customers affected by an interruption. Different customer categories are represented by average cost rates that aim to represent the different end-user consequences (G. Kjølle et al., 2008). The CENS cost related to a power outage represents the interruption's societal costs but does not consider other costs than the end-user's. For example, the CENS rates do not account for the cost of unavailable public services (G. H. Kjølle et al., 2012).

2.2 The utility mast

This section presents the building blocks that make up a typical Norwegian utility mast and their relevance to reliable power delivery. Furthermore, we show how a utility mast wears out over time, by the deterioration of its components.

2.2.1 Utility mast composition

Utility masts are connected by a transmission line and make up a *utility line*. They transfer power from one location to another. A utility mast can be the part of either the transmission-, regional or distribution grid. While utility lines vary in length and composition, their underlying structure is similar.

Most of Norway's utility lines are overhead, opposed to underground, and overhead utility masts will be the focus of this thesis. They hold up the utility lines to reduce interference from humans, animals and vegetation close to the ground.

A collection of *components* with different characteristics and functions, makes up a utility mast. Some components keep the transmission line is at a regulatory height and sufficient distance from other objects (Augland and Staurvik, 2014), while others directly influence the mats' ability to deliver electricity.

Figure 2.1 illustrates a typical Norwegian utility mast. The pole and traverse are of either wood or steel. With large forests in Norway, wooden poles are common cost-effective, yet sturdy alternatives in the regional- and distribution grids. For the transmission grid, steel is more common. Steel allows higher, more solid structures. It is also safer as the lines with high voltage become less accessible from the ground, and the masts are less prone to sudden failure. The foundation and backstay support the pole and are also essential for the pole to be reliable. Some components secure a stable and safe electricity transfer. The insulator, spark gap, and deflector are examples of such components. The transmission line is not a part of the utility mast itself, but the section that is close to the mast is often considered a utility mast component when describing it.

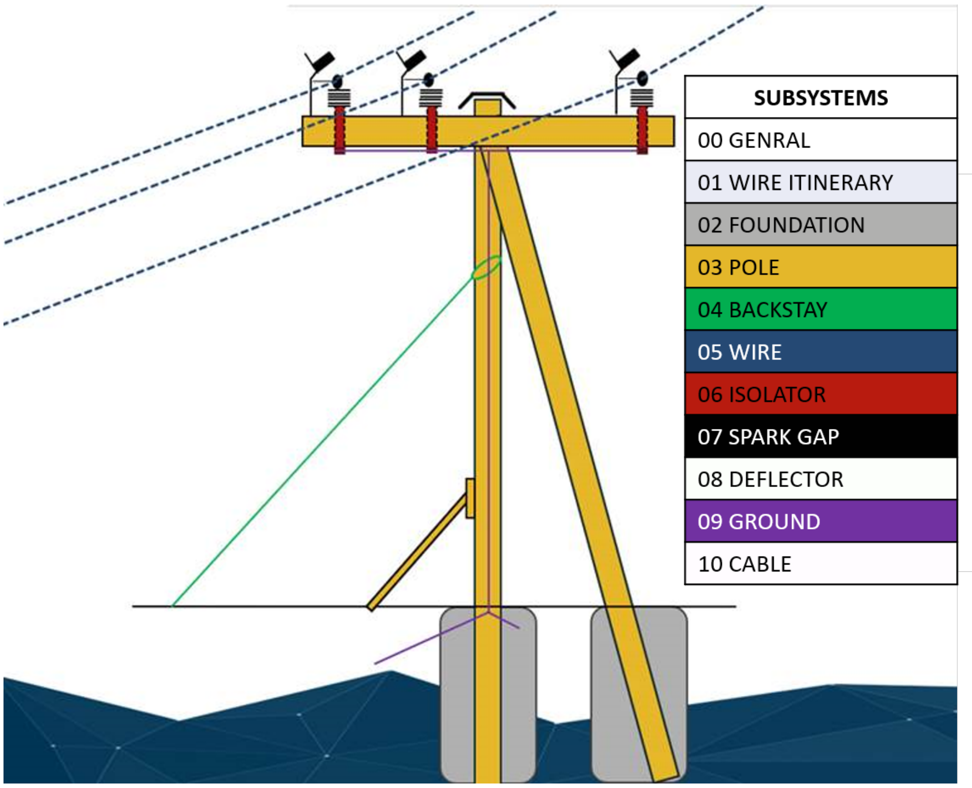


Figure 2.1: A visualization of a typical Norwegian utility mast, provided by Wiseline AS

Across the Norwegian power grid, masts are exposed to different environments in terms of weather, constructions and terrain. Masts also have different purposes as to the type of transferred electricity, and high-voltage transmission requires different mast structures than low-voltage transmission does (Riibe and Weyergang-Nielsen, 2010). In this thesis, we study a typical Norwegian utility mast and seek to formulate a general model than can be applied to different masts given the right component data. For a complete description of overhead utility mast components, we refer the reader to (Nordnes, 2011).

2.2.2 Deterioration

When a system's condition falls from a higher, better level to a lower, worse level, the system *deteriorates* (Nicolai, 2008). All utility mast components will worsen over time and eventually fail. Depending on the component, it may be prone to sudden shocks that drastically worsen its condition. Other components may gradually wear out over time. A combination of these two types of deterioration is also possible. A utility pole may continuously degrade due to rust or rot over several years before an external shock such as a lightning strike hits it. Depending on the shock's severity and the degree of degradation that has happened before it, the utility mast may fail to deliver electricity.

Deterioration may vary significantly across a period of time as well. Woodpecker holes contribute to faster deterioration of wooden utility poles, but the number of woodpeckers that visit a utility pole in a year can differ from year to year. Some components may be more prone to immediate failure than continuous deterioration. Trees falling over utility lines is a typical cause of electricity outage (Solvang and Foros, 2019).

Although it is impossible to say precisely what will happen to a utility mast in the future, it is possible to estimate its components' expected deterioration. By doing so, one will find the *expected lifespan* of each component that comprises the utility mast. Typically, the expected lifespan is based on a component's nominal load and empirical data on how external factors have influenced it earlier. For a utility mast, the expected lifespan of its components is given in years. The number of years that a component carries out its intended task is called the component's *lifetime* (Stene et al., 2005).

If a component's function directly influences a masts' ability to deliver electricity, it is a *critical* component. A utility mast has both critical and non-critical components. A non-critical component failure will be the sole cause of interruption in power delivery. However, they may influence the deterioration process of other mast components. Removing the top hat from a utility mast does not cause a power outage, but it will make the pole rot faster. Several non-critical components' failure may also cause failure in power delivery as the utility mast may need a sufficient number of functioning non-critical components to deliver electricity.

2.3 Maintenance

This section focuses on what maintenance is and how it can be utilized for better asset management. We present the structure of maintenance decision making and introduce the research field that is maintenance optimization.

2.3.1 The purpose of maintenance

Throughout this thesis, *maintenance* will refer to a definition from The United States Departement of Defence (United States Departement of Defence, 2019): Work that allows a system to carry out its intended task by improving its condition. The term *repair* specifies restoration of a system *not* carrying out its indented task, as maintenance may be either a repair or carried out to prolong a systems lifetime. Both repairs and general maintenance may involve *replacements* of system components.

Kuo et al. (2001) includes two types of maintenance when describing accepted principles for increasing system reliability. The first is repair maintenance, meaning manual replacement of forces when they fail. The second is preventive maintenance, meaning replacing forces either when they fail to carry out their intended task or at some fixed interval if they have not failed yet.

In the survey by H. Wang (2002); maintenance is categorized into two different classes, *Preventive maintenance* (PM) and *Corrective maintenance* (CM). These terms are quite aligned with the above definitions, where repair and corrective maintenance are subject to the almost same interpretation. An important distinction is that H. Wang (2002) introduces two adjunct concepts, where actions on a failed force never fall into the PM-category. The survey, reviewing a large amount of literature regarding maintenance, also includes repairs in addition to replacements. PM is carried out on an operating system with operating forces, and CM when a failure has occurred.

Throughout this thesis, the focus will be on preventive maintenance based on the desire to restore a system's function, should it, at some point, fail to carry out its intended task. Preventive maintenance includes inspection decisions in order to obtain information and make optimal choices. Corrective maintenance is a result of failure and is rarely desirable, but maybe accepted at some risk level due to economic benefits of a lover frequency of repairs and replacement. This risk level is in close relation to the systems *reliability*. How often and when to carry out PM versus CM, is a result of the *maintenance strategy*.

2.3.2 Maintenance decision making

A maintenance strategy aids decisions regarding the type, timing and frequency of maintenance (Muchiri et al., 2011). Pintelon and van Puyvelde (2006) explains *maintenance decision making* by dividing it into three. A maintenance *action* is the basic elementary work needed on a component. A maintenance *policy* is the rules that describe which mechanisms trigger different maintenance actions. The structure which policies and actions are based upon is a maintenance *concept*. Reliability centred maintenance (RCM) and Total Productive Maintenance (TPM) are two concepts where the former generally focuses on the risk of system failure, while the latter takes an organization-wide approach for avoiding failure and quality assurance. The terms "strategies" and "concepts" are often used interchangeably (Nakajima, 1988).

According to Lam and Yeh (1994), a maintenance *policy* reduces total costs and avoids failure of a system. We see here a broader interpretation of this term. Throughout this thesis, we will use the same definition as recent literature reviews that try to clarify the use of these terms. Maintenance concepts may include qualitative decisions, while maintenance policies only base decisions on measurable parameters (Sharma et al., 2011). A Maintenance policy may be age-based, time-based or condition-based, depending on which parameters are measured and trigger maintenance actions. A policy may be based on several parameters, and even the binary parameter stating whether the systems work or not. A policy relying only on the latter is referred to as run to failure (RTF) policy (van Horenbeek et al., 2010).

2.3.3 Maintenance optimization

Maintenance optimization emerged as a research discipline after the Second World War. This was due to the increased acknowledgement of how maintenance planning is important for cost-efficient asset management (Ben-Daya et al., 2016). The discipline includes the use of mathematical models to predict when items fail. This use highlights how maintenance is also important for *preventing* these failures and repairing those items which have already failed. An appropriate definition of maintenance optimization is:

"*A method aimed at determining the most effective and efficient maintenance plan (i.e., inspection time and frequency, work preparation, required maintenance resources) so that the best possible balance between direct maintenance costs (e.g. manpower cost, logistics and transportation costs) and the consequences of not performing maintenance (e.g. loss of power production and assets) is achieved.*" (Shafiee and Sørensen, 2019)

Maintenance optimization became widely recognized in the 1960s because of its evident use for preventing failures and unplanned downtime. This use meant lower cost related to asset management. This cost was minimized through the use of different Operation Research (OR) models (Pintelon and Gelders, 1992) to make *maintenance optimization models*. In this thesis, we will focus on such models. This focus excludes several OR-models used concerning maintenance, such as models for inventory control or project management.

Note that, from the definition, maintenance optimization includes the optimization of an inspection plan. This thesis is particularly concerned with inspections role in maintenance optimization, and the reader will thus find separate sections dedicated to it. However, in this thesis, the term "maintenance optimization" includes optimizing inspection decisions unless otherwise stated. Today, maintenance optimization is commonly used on systems, as companies consider this a profit-generating business element (Kutucuoglu et al., 2001). Consequently, the potential for maintenance optimization research is of general interest.

Generally, we struggle to find applications of maintenance optimization to power grid maintenance. Although operators focus on effective maintenance decisions that are cost-effective, they are less concerned with finding the *optimal* choices. The challenge of applying maintenance optimization models to specific problems is acknowledged within the field and is not only related to power grid operation (van Horenbeek et al., 2010). Consequently, the Norwegian power grid operators do not focus much on optimizing their maintenance and inspection decisions (Energi Norge, 2017). In 2019, the cost of operating and maintaining the Norwegian power grid was a total of 9.6 billion NOK. This cost implies significant economic potential in finding the optimal choices related to power grid maintenance.

2.4 Inspection

When faced with the decision of maintaining a system, there are several factors to take into consideration. Although minimizing expected costs means optimizing the balance between costs and benefits of the available decisions, deriving this optimum can be far from trivial. Forecasting a system's development through deterioration models and how maintenance actions affect the deterioration, is of great aid in making the decisions. Still, if one is to decide on maintenance, one must also have an idea of what condition the system is in at the time of the decision. This section presents how inspections are considered in relation to maintenance optimization before introducing typical categorizations of inspection decision structures in maintenance optimization problems.

2.4.1 Inspection and its relation to maintenance optimization

Within all industries, maintenance planning includes deciding maintenance methods and choosing inspection frequency (Verma et al., 2006). Most of the maintenance optimization models allow taking a look to know or estimate a system's condition. Some even require it before carrying out any maintenance (B. Liu et al., 2017). *Inspection* is a way to acquire this information and plays an important role in maintenance optimization.

Onoufriou and Frangopol (2002) state that inspection is a way of ensuring structures' safety and serviceability, but point out that it can represent a high cost. The role of inspection has evolved from being based on general guidelines and judgement, to optimize future actions when incorporated in a planning and decision model. Thomas et al. (1991) summarize how inspection relates to maintenance optimization problems:

"*Inspection involves examining deteriorating systems to try to identify their state, in order to effect some repair, replacement, or maintenance action.*" (Thomas et al., 1991, p. 283)

Modelling inspection is an integral part of modelling a maintenance optimization problem. Inspections may differ in cost, the kind of information they return and the accuracy of this information. Inspection can be modelled differently in terms of costs, information accuracy and type of information. The decision space in the problems discussed so far has varied from merely deciding whether to inspect or not, to several possible inspection types.

2.4.2 Inspection structures in maintenance optimization

Inspection can be considered as either an uncontrollable source of information, a negligible part of a maintenance optimization or as performing a set of actions. These three options represent different degrees of inspection *flexibility* (Durango-Cohen and Madanat, 2008). Additionally, the inspection can be completely *controllable* or restricted to some rules such as is the case for inspection of the Norwegian power grid. Although the power grid operators can choose when to inspect their grid thoroughly, they are forced to do so at least every tenth year. Furthermore, their set of inspection actions contain one that gives superficial information about the utility lines and one that is more thorough.

If the inspection is controllable, a maintenance optimization problem may require a specific *inspection policy structure*. According to Nakagawa and Mizutani (2009), there are three types of structures:

- *Periodic inspection*. Inspections are performed at periodic times. Before conducting a maintenance and inspection plan, all inspections are planned and distributed evenly across the time specified in the plan. Thus, one performs inspection periodically, at pre-determined, even intervals.
- *Sequential inspection*. Also called *aperiodic inspection* because the condition of even intervals is removed, meaning that one can regard the inspection decisions as sequential. In every stage where an inspection is performed, one decides when the next inspection is, independently of earlier inspection decisions.
- *Asymptotic inspection*. Involves minimizing the expected cost of system failure by finding an appropriate interval with the optimal probability of system failure. One carries out inspections at these intervals.

The last structure type, asymptotic inspection has the same structure as periodic inspection, but is derived differently. Therefore, one may therefore subdivide inspection policy structures by only periodic and sequential inspection (de Jonge and Scarf, 2020). This thesis focuses on periodic and sequential inspection. We derive two models that return an optimal periodic and sequential inspection policy, respectively.

Sequential inspection is sometimes referred to as *dynamic inspection scheduling*. This emphasizes the fact that sequential inspections allow decisions to be made for the next interval in question and does not require pre-determination such as the periodic inspection. Sequential inspection policies are well known to be advantageous compared to periodic inspection policies (Verma et al., 2006).

2.5 Maintenance and inspection of the Norwegian power grid

In Norway, harsh weather and severe storms commonly cause transmission line failure. The reason for a blackout is often intense periods of wind and icing, or sudden lightning strikes (Kiel and G. H. Kjølle, 2019). Combined with utility masts' natural deterioration processes, the Norwegian power grid operators face several threats to reliable and cost-effective power delivery.

Inspection and maintenance costs largely contribute to the operating costs related to power grid management. To ensure cost-effective life cycles for the utility masts, power grid operators need to plan both inspection and maintenance of the grid. This planning involves deciding when to send technicians to inspect the power grid line and what to do with the utility masts based on the information obtained from inspection. An inspection of a utility mast returns information about the mast components' condition. Additionally, dangerous situations may arise near the utility masts. This is a natural consequence of transportation and supply of electricity. Power grid line inspections can detect these situations and ensure appropriate action for resolving them (Nordlandsnett AS, 2019).

Technicians inspect the utility line either from the ground or by helicopter. Different *types* of inspection require different equipment and some types are more comprehensive than others. Consequently, the condition of utility mast components is returned with different degrees of certainty. A visual inspection is typically less accurate than analyzing components in a laboratory. Technicians may also use measurement tools specifically created for the inspection of some utility mast components (Nordnes, 2011). Furthermore, the different inspection types will require different amounts of time and have different costs associated with the inspection.

The grid operators apply an inspection and maintenance strategy on their grid to balance maintenance and inspection costs with an acceptable risk level concerning interruption costs, safety, and reliable power supply. The operators find an appropriate periodic interval and a maintenance plan, basing their decisions on risk analysis, and the condition returned from inspections. Maintenance decisions involve repairs or replacements of utility mast components (Wiseline AS, 2017).

The most common maintenance policy applied by Norwegian power grid operators today is *condition-based maintenance (CBM)*. This policy means that when an inspection reveals a mast component condition that exceeds a certain threshold, they repair or replace it (B. A. Ellis, 2008). Often, the individual components' thresholds may be set based on the technician's expertise or empirical data.

Finding the optimal threshold for each component can be challenging. Furthermore, as some costs are associated with a technician only travelling out to a utility mast, it may be cost-efficient to repair less deteriorated components if *some* component condition exceeds its threshold.

As the cost of operating the Norwegian power grid influences the price of using the grid for customers, more efficient inspection and maintenance have socioeconomic value. With efficient strategies, cost of maintenance and re-investment will be lower, making optimization of the Norwegian power grid maintenance and inspection an attractive field of research.

Illustration of maintenance and inspection decisions on a utility mast

We conclude this chapter by illustrating how a utility mast may deteriorate and the choices and information related to such a process. We assume that the power grid operator can only choose one inspection type to gain information about the mast's condition. Immediately after an inspection, the power grid operator must also choose between maintaining the whole mast or leaving the mast in its current condition. Furthermore, we assume that a system failure is self-announcing. This means that if the mast fails, the operator will immediately know, possibly because of a blackout.

Figure 2.2 describes the whole process. The blue line indicates the actual condition of a mast, which is unknown to the operator. As time passes from left to right in the figure, the mast will deteriorate and be closer to failure. However, the operator knows that the mast deteriorates in a probabilistic manner and can therefore be certain of an interval representing the mast's possible condition. The grey areas in the figure mark this *belief*. A red plus indicates an inspection and a green circle indicates maintenance.

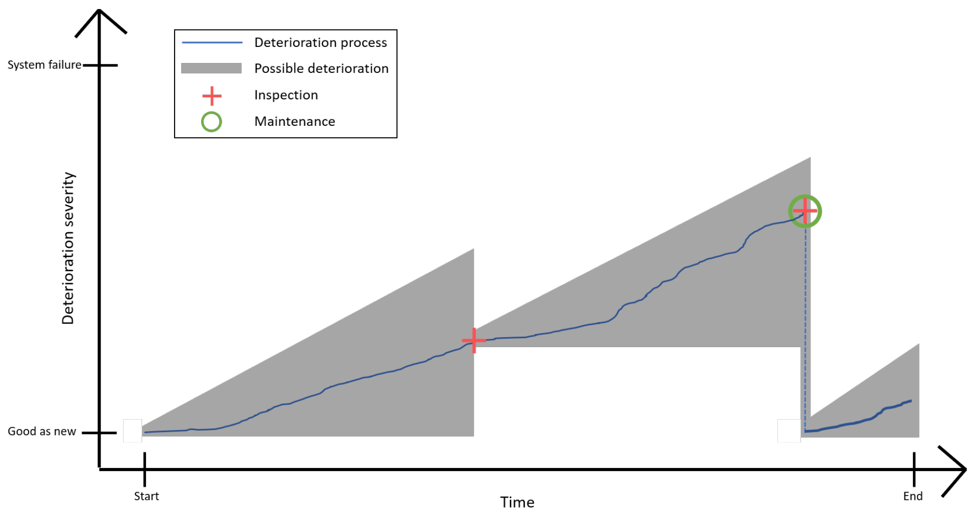


Figure 2.2: The figure shows an illustrative example of how a utility mast may deteriorate over time, and the information and decisions available to a power grid operator.

We see that the mast is known to be "good as new" at the start of the period, and gradually deteriorates. While deteriorating, the interval representing the operator's belief of the mast condition becomes larger. This increase is natural, as the deterioration process may have developed in several different ways and these possibilities increase over time. Note that the operator knows that the mast has not failed although it may do so because of a falling tree or a lightning strike. This knowledge is because such an event would immediately make itself known to the operator.

When the operator first decides to inspect the mast, the inspection reveals the actual condition and the operator decides not to maintain the mast. The belief interval is then much smaller, and the operator again waits before the next inspection. At the next inspection, the mast's revealed condition has adequately deteriorated, and thus the operator decides to maintain it. The maintenance in this example sets the mast's condition back to "good as new", and the operators takes no further action in the remaining time illustrated by the figure.

The illustrative example explains the problem faced by power grid operators. They can inspect their masts in order to obtain information about their condition. Furthermore, they can choose to maintain a system before it reaches the failure condition. An operator's choice of inspection and maintenance of a mast is subject to the answer to an important question: Is it more economical to inspect now and possibly replace the system to avoid it reaching the failure state? Kao (1973) formulates a variation of this question. If the answer is positive, an operator inspects the mast and asks the same question before maintaining it.

Chapter 3

Related Literature

This chapter presents maintenance optimization literature focusing on how inspection is modelled in maintenance optimization problems. Furthermore, we introduce the reader to the theory behind solution approaches used in this thesis, supplemented with illustrative examples and similar approaches within the field.

Section 3.1 provides historical context as to how maintenance optimization models have evolved since the field's birth. Section 3.2 introduces common terminology and gives an overview of maintenance optimization literature. Inspection is superficially mentioned in relation to the literature discussed Section 3.2 as it is more thoroughly discussed in Section 3.3. We dedicate an entire section to inspection in maintenance optimization because of its importance to this thesis. Section 3.4 considers theory about the Markov decision process and highlights this process's use in maintenance optimization literature. The last section, 3.5, introduces dynamic programming, a common solution approach to maintenance optimization problems. The last section also discusses some relevant publications that make use of a dynamic programming approach.

3.1 The evolution of maintenance optimization models

Maintenance optimization has received wide attention among researchers since the 1940s. A model of a maintenance optimization problem includes many factors. Consequently, one finds many attempts at classifying models based on their characteristics. These attempts include the work of Barlow and Proschan (1965), Sherif and Smith (1981), Dekker et al. (1997) and Sharma et al. (2011). This section provides an overview of how maintenance optimization models have evolved and thus motivate the characteristics used to classify these models today.

Attempts at classifying maintenance optimization models based on common problem characteristics date back to the 1960s. McCall (1965) categorizes such models based on the available information about a system's distribution of times to failure. In the same year, Barlow and Proschan (1965) categorized maintenance optimization models by how decisions were made. They differentiate between continuous-time and discrete-time decisions.

Thomas (1986) introduces his survey by claiming that the literature on maintenance recently has shifted to consider systems comprised of several building blocks, from traditionally considering singe-item systems or components. The maintenance optimization literature from the 1980s typically model systems comprised of several items. This indicates that there are important distinctions between models that consider systems as indivisible and those who consider several building blocks making up a system. The shift is a consequence of better techniques for analyzing complex systems. Earlier research also highlights the importance of interactions between units in a system for the system's reliability.

Thomas et al. (1991) classify maintenance models of deteriorating systems by four facets: The stochastic description of how the system deteriorates, how the system can be improved through available maintenance actions, the criterion to be optimized and the availability of information about the systems' state. The latter facet is closely related to the information acquisition of a system's state, possibly modelled as a choice rather than just modelling the available information as fixed. Thomas et al. refer to the information acquisition of a system's condition that aids maintenance decisions as *inspection*.

Dekker (1996) defines maintenance optimization as optimizing the balance between cost and benefits of maintenance. This definition implies that a maintenance optimization model requires quantification of these said benefits and costs. In their review of maintenance optimization articles from 2001-2018, de Jonge and Scarf (2020) state that maintenance optimization includes both analysis and development of mathematical models. These models aim at optimizing or improving maintenance policies. This definition is broader and does not require any specific quantification. Regardless, optimization naturally leads to the quantification of some parameter one aims to optimize. The literature considers different *optimality criteria*, such as minimizing the cost rate or the total costs within a specific time period. Another optimality criterion is the maximization of availability or reliability. Y. Wang and Pham (2011) optimize for both costs and availability (by minimizing costs and the unavailability).

General maintenance optimization models provide theoretical insight into how mathematical programming can aid maintenance and inspection decisions. However, they have limited impact on actual maintenance management because of the inadequate problem definitions provided by maintenance modellers (Sharma et al., 2011). This both highlights the importance and difficulty related to accurate mathematical representations of real maintenance and inspection problems. Application of academic models on specific business problems is challenging. The gap between theoretical models and their applicability for actual decision support is claimed to be the biggest problem within the field (van Horenbeek et al., 2010).

3.2 Overview of literature on maintenance optimization

The purpose of this section is to introduce the reader to the existing literature on maintenance optimization and discuss the general contents of a maintenance optimization model through examples from earlier publications. We highlight important aspects of the research field by describing the problems modelled in relevant publications. This section begins with a presentation of relevant terminology, before discussing publications with different modelling approaches. Following the chronological development of research within the field, we first consider the modelling of single-unit systems before using established standard features of these models to introduce the increased complexity of modelling multi-unit systems. Then, we discuss deterioration modelling in the literature before describing some optimization techniques used to solve maintenance optimization problems.

3.2.1 Common terminology used in the literature

For a complete discussion of terminology used in the literature for modelling, managing and optimizing maintenance, we refer the to Ben-Daya et al. (2016). Their discussion begins by introducing the term *engineered objects*, exemplified by single products or complete infrastructures. Engineered objects are claimed to be unreliable and therefore, in need of maintenance. Such objects are commonly referred to as *systems* in the literature. A system can be considered as an asset performing an operational function (de Jonge and Scarf, 2020). The asset deteriorates and is thus subject to maintenance.

A system may have a hierarchical structure, consisting of *sub-systems*. These sub-systems inherit the characteristic of a system, and may therefore also consist of additional sub-systems. On the lowest level in the hierarchy, the sub-systems cannot be divided further into sub-systems. These indivisible systems are called *components*. Depending on the maintenance optimization model, a component's ability to function may be described by a *state*, ranging from fully functional to failed. The possible states that a component or system may be in, the *state space*, can be either a discrete set of states or a continuous interval representing the aforementioned range (Y. Liu et al., 2020).

The literature uses the terms "*condition*" and "state" interchangeably. Therefore, it is difficult to distinguish the use of these two terms completely. Typically, "condition" is used in a general context, e.g. when describing a problem at a higher level. The introduction of a mathematical model in a maintenance optimization problem often includes representing a system's condition by states in a state space. Thus, "state" is generally used when discussing the model. An example is the model proposed by W. Wang (2007), where the term "condition" frequently occurs in the introduction, but never when introducing notation and formulating the model, as opposed to the term "state".

Maintenance optimization models regard a system either as a *single-unit*/*single-component* or *multi-unit*/*multi-component* system (H. Wang, 2002). Because of the possible hierarchical structures with several levels of sub-systems within a system, we the term "multi-unit system" throughout this thesis for systems that may be further divisible into sub-systems. When the term "component" is used (instead of "sub-system"), it is to emphasize that there are no lower levels in a system model.

3.2.2 Single-unit system models

A maintenance problem considering only a single component was first described by Derman (1963). The article considers a component whose state is revealed through inspections at intervals set before the maintenance decisions. The system deteriorates from a "new" state into a "final" state. The cost of replacing the component increases if the system fails, making it optimal to replace the component right before failure if one was to have complete information about the component's state at all times. However, this information was only made available at the mentioned inspection intervals. When seeking to minimize the cost of maintenance, one faces a trade-off between lower risk of costly failure, and the extra cost related to a possibly avoidable replacement. In the article, the objective is to provide a condition-based rule, stating that the component should be replaced if it reaches a certain state (or a worse state) at the time of an inspection. A Markov Chain with stationary transition probabilities describes the deterioration process. We further elaborate on this particular description in Section 3.4.

The single-unit model from Derman (1963) was since its publication subject to several expansions in later literature. Pierskalla and Voelker (1976) refer to extensions made shortly after the first publication of a single-unit model, in an early survey of maintenance models. Kolesar (1966) preserve the optimality of the initial solution but extend the model to regard a non-decreasing "occupancy" cost for increasingly deteriorated conditions. The occupancy cost can be interpreted as an increasing cost for maintaining the system as its state worsens.

citetross1969markovian, introduce a more general state space, allowing a continuous state-development earlier modelled by a set of discrete steps. Kao (1973) proved results similar to the works above by allowing randomization of inspection intervals, meaning that the information reveal was not made based on pre-determined intervals. However, it is important to notice the difference between random inspection and *dynamic inspection decisions*. The latter allows the decision maker to reveal information at different, not necessarily evenly, distributed time points across the total time considered in a model. As Pierskalla and Voelker (1976) published their survey quite shortly after the single-unit model's introduction, several later extensions were not considered there. Sherif and Smith (1981) mention some of these extensions is their survey, showing that common aspects of modern maintenance models were quickly taken into account by later publications.

Although the literature on maintenance optimization has shifted towards multi-unit systems, these early publications introduce common aspects of modern-day maintenance modelling and prove useful for describing multi-unit models. Research on single-unit systems is still being conducted. More recent publications include a global approach by Chu et al. (1998), considering a continuous state space where the information about a component's state is uncertain at the time of a maintenance decision. This approach is distinctively different from the other discussed work, where revealed information is assumed to be certain. Berrade et al. (2015) consider failures that are not self-announcing. That is, a failure is only detected by an inspection. The objective is to decide the optimal interval for carrying out these inspections. Cha et al. (2017) model how the probability of a system *shock*, a sudden damage, may increase with its age. The paper provides an interesting discussion on a shock's double effect, as it may affect an item's state and the probability of further deterioration. A component's probabilities of deteriorating from one state to another are commonly modelled for each possible state transition and referred to as the *transition probabilities* (van Oosterom et al., 2017).

Today, publications on maintenance optimization considering single-unit systems often address specific characteristics of a model and suggest changes that differ from the most common model adoptions. These changes often focus on specific extensions that better relate to specific real-life industrial business cases (de Jonge and Scarf, 2020). This approach also coincides with the challenge suggested by van Horenbeek et al. (2010), addressing a lack of applicability for general maintenance optimization models. Different models have different associated assumptions regarding state representation, maintenance possibilities, and information inspection returns.

3.2.3 Multi-unit system models

With the shift in maintenance optimization literature from considering only one unit to systems comprised of multiple components, the interaction between sub-systems naturally received greater interest. It is common to classify this interaction between units into three types (Dekker et al., 1997; Nicolai and Dekker, 2008; Laggoune et al., 2010; Shen, Hu et al., 2020). These three types are:

- *Economic dependence:* Sub-systems are economically dependant if the total cost of maintaining them one by one is different from the cost of repairing all of these sub-systems at once. Zhou et al. (2016) demonstrate this by modelling a pump system consisting of a motor and a pump. When repaired simultaneously, the cost of setup and productivity loss may be reduced compared to repairing the sub-systems individually at different times.
- *Structural dependence:* Sub-systems are structurally dependant if maintaining one sub-system implies a maintenance action on another sub-system, such as dismantling, replacing or regular maintenance (Dao and Zuo, 2015). Dinh et al. (2020) illustrate structural dependence using a gearbox system. In order to remove a specific gear, a specific bearing needs disassembling. S. Wu et al. (2016) model another type of structural dependence. In their model, some sub-systems' failures invoke the possibility of maintaining other sub-systems during the system's downtime.
- *Probabilistic dependence:* Also referred to as *stochastic dependence* (Thomas, 1986; Shen, Hu et al., 2020). Probabilistic dependence between sub-systems means that their state influences one or several other sub-systems' lifetime distribution or one or several other sub-systems' state(s) influence their lifetime distribution (Shen, Elwany et al., 2018). When introduced, only sub-system failure affecting the probability of other sub-system failures were considered (Murthy and Nguyen, 1985). This type of dependence has later been extended to include all parts of the degradation process, not only the processes going directly to failure (Gao et al., 2019). Li et al. (2016) propose a model with the latter mentioned, more general probabilistic dependency in a system with two components.

Other examples of dependency-centred literature include introducing a shared setup cost for maintenance by Lugtigheid et al. (2008). A binary maintenance decision for each sub-system incurs a setup cost when maintaining *any* sub-system, and does not change with respect to the number of maintained sub-systems at once. They represent the system's state by a weighted sum of sub-system-states, assigning weights according to how critical a sub-system is for the system. Another approach addresses the possibility of carrying out maintenance on other sub-systems when a sub-system failure *forces* the fixed setup cost, adding a decision to an event traditionally regarded deterministic (one only fixes the failed sub-system and wait for the next opportunity to carry out regular maintenance).

In the literature on multi-unit models, economic dependencies are the most commonly addressed type of dependency (de Jonge and Scarf, 2020). However, the other types of dependencies are also frequently discussed. Examples given with the definition of each dependency above substantiates this claim. By modelling several machines and limiting the maintenance resources, Armstrong (2002) addresses *resource dependency*, traditionally considered as another type of economic dependency (Thomas, 1986). The machines must be shut down during maintenance, making it costly to plan for simultaneous maintenance if the number of idle technicians is lower than the number of machines to be maintained.

Keizer et al. (2017) claim that recent literature development makes the classification mentioned above insufficient because of the increased interest in resource dependency. Although the paper is widely cited, we fail to find a general acceptance within maintenance optimization literature for this adoption. In their publication from 2020, Shen, Hu et al. refers to the traditional three types as "common to see in the literature".

Recent publications on maintenance optimization of multi-unit systems commonly delve into specific aspects of general maintenance problems. As a natural consequence of considering systems composed of multiple units, many recent papers focus on sub-systems' dependencies. Models also tend to address deterioration processes differently, especially considering how state spaces are defined.

3.2.4 Modelling deterioration

One carries out maintenance in order to keep equipment operational, yielding the required output and quality. Maximizing the equipment's ability to do so in a cost-effective way is considered the objective of maintenance (Pintelon and Gelders, 1992). As systems are subject to deterioration, maintenance is unavoidable (Nicolai, 2008). Consequently, modelling how a system deteriorates is essential for modelling and optimizing maintenance decisions. The single-unit system modelled by Derman (1963) worsens gradually with time, implying that the component's state distribution is dependant on its age. Deterioration of a system is generally considered a result of ageing, usage or fatal shocks (Chiang and Yuan, 2001). The deterioration is commonly considered uncertain, e.g. through random shocks on the system (Y. Wang and Pham, 2011) or due to different possibilities for state transition with ageing (Maillart, 2006). This type of deterioration is called *stochastic deterioration* (McCall, 1965).

Maintenance optimization models were early on categorized based on whether decisions were made at discrete time points, or continuously (Barlow and Proschan, 1965). These categories may also be used for the state space and consequently for the deterioration processes (Ross, 1969). The model we suggest in this thesis has a discrete structure for both decisions and states. A common approach resulting from these assumptions is modelling a *Markov process* to benefit from the discrete structure regarding the uncertain state of a system (Byon and Ding, 2010). Section 3.4 elaborates the characteristics, implications and modelling benefits of these processes.

The assumption of discrete state space and deterioration may be unreasonable for some systems. Some processes can describe these spaces as continuous. Alaswad and Xiang (2017) describe processes used when modelling continuous-state deterioration. Liao et al. (2006) suggest a gamma process to model continuous deterioration, while X. Liu et al. (2013) use a geometric Brownian process. For additional literature on continuous-state deterioration modelling, the reader referred to Alaswad and Xiang (2017). The article refers to different modelling approaches, including the use of an inverse Gaussian process. As both the model and solution approach in this thesis consider a discrete state space, the remainder of this chapter will focus on how the literature exploits this chosen approach.

3.2.5 Optimization techniques

A Markov deteriorating system and the decision structure of maintenance optimization problems are commonly exploited by modelling a *Markov Decision Process* (MDP) (Alaswad and Xiang, 2017). Such a decision process can be defined by a set of discrete states, transition probabilities, a set of possible actions and a cost distribution related to decisions and/or events (Levin et al., 1998). Problems formulated as MDPs are widely solved by using *dynamic programs* (Alaswad and Xiang, 2017). We discuss and illustrate MDPs in Section 3.4 and this solution approach in Section 3.5.

The literature on optimization techniques for solving maintenance optimization problems naturally extends beyond dynamic programming. We refer the reader to van Horenbeek et al. (2010) for an overview of optimization algorithms used in maintenance optimization problems. Different optimization algorithms are both classes in the provided classification framework and the topic for a short discussion with further references to research applying these algorithms. The article discusses traditional algorithms such as linear and integer programming alongside metaheuristics, evolutionary algorithms and those considering multiple objectives. As we use dynamic programming as the solution approach to a maintenance optimization problem, we consider a further discussion of other optimization techniques beyond this thesis's scope.

3.3 The role of inspection in maintenance optimization

This section introduces the role of inspections in maintenance optimization problems. We consider several modelling approaches using inspection decisions to optimize a maintenance problem. First, the modelling choices related to cost and accuracy of inspection are discussed and exemplified through publications in Section 3.3.1. Then, we introduce the different inspection types relevant for a maintenance optimization problem in Section 3.3.2. We also discuss relevant literature of the two most commonly considered inspection decision structures introduced in Chapter 2. These are called "periodic" and "sequential" inspection, and are presented in Section 3.3.4 and Section 3.3.3, respectively. Throughout the different parts of this section, we present publications considering single-unit systems before publications on multi-unit systems. We see that especially one type of inspection policy structure, sequential inspection, is little researched for multi-unit systems.

3.3.1 The cost and accuracy of inspection

Thomas et al. (1991) introduce a categorization of inspection models based on the cost of acquiring information and how accurate this information is. For each of these two factors, they consider two different possibilities. The information is either *costless* or *costly*, meaning that information either is acquired at a price or is freely available. Information can also either be *perfect* or *partial*. The former means that the system's actual state can be acquired, while the latter implies uncertainty about this state, possibly represented as a probability distribution over several states. By combining these factors, they categorize models based on the price (that may be zero) and availability (or unavailability) of both partial and perfect information.

The most straightforward approach taken to include inspection in a maintenance optimization model is assuming continuous monitoring of a system, at no cost. Thus, perfect information is available for free. Kawai (1983) conducts such an approach by modelling a system that requires an inspection before maintenance. Inspections are assumed to be costless and provide perfect information, making the decision of inspection irrelevant. They are irrelevant because the model can assume that perfect information is always available, as the model neglects the cost and uncertainty of information. Assaf and Shanthikumar (1987) also assume perfect and free information in their model considering the problem of maintaining a group of machines. Although they do not require inspections to carry out maintenance, the objective function value is not affected by when and how inspections are conducted, making it optimal to obtain information as often as possible. Again, the assumption of free and perfect information makes inspection decisions irrelevant, reducing the problem to just finding the optimal maintenance policy.

In reality, inspections rarely provide perfect state information. Consequently, this assumption does not apply to all systems. Durango-Cohen and Madanat (2008) claim that maintenance optimization models commonly assume perfect information from inspection because it enables modelling with Markov decision processes (which we discuss in Section 3.4). These models implicitly assume perfect information and exact deterioration probabilities. Especially the former assumption has received attention. If the assumption of perfect inspections does not hold for a specific model, it is common to represent an inspection's return as a probability distribution of possible states (Byon and Ding, 2010; Andriotis and Papakonstantinou, 2020; Morato et al., 2020).

Papakonstantinou and Shinozuka (2014) suggest several possible approaches not subject to the limitation of assuming perfect information. These approaches include the partially observable Markov decision process (POMDP) which we discuss in 3.4.5. Approximation methods collectively referred to as point-based solvers are highlighted as a promising approach to such process models. Point-based solvers use a lower bound initialization to update estimations of the objective value, and iteratively seek to improve the accuracy of both the system's state and the objective value.

Although the approach with point-based solvers is exciting and reports new results (Morato et al., 2020), the assumption of perfect information is also commonly made because of its relation to many existing problems.

The combination of available technology and methods for inspection can often provide excellent information about a system. Especially for discrete state spaces, assuming that an inspection will correctly provide a system's state is reasonable for many problems. In the paper on underground cable maintenance by Bloom et al. (2006), the authors address that the diagnostic tests (a form of inspection) are inaccurate. However, they state that an inaccurate test can be useful. Their model assumes perfect information, and as the state space is discrete, one can assume that the tests provide correct classification, although the reality includes imperfect information.

3.3.2 Types of inspection

In more recent literature, the opportunity that lies in controlling inspections has received greater attention. One option is to model how hidden failures can be detected often enough to avoid larger costs at a later time (Berrade et al., 2015). Furthermore, B. Liu et al. (2017) exploit inspection as a fragmented decision. They consider inspection as an individual decision for each sub-system with costs being partly fixed for any inspection at all with a variable cost assigned to each kind of inspection. The term *inspection types* is used in the literature to denote the fragmentation, where an inspection type can include either a set of sub-systems, the whole system, or single sub-systems.

H. Ellis et al. (1995) model a bridge subject to a possible inspection every other year, and maintenance every year. Two types of inspection are carried out on the bridge: A visual or ultrasonic inspection. Not carrying out an inspection is also allowed, but we do not refer to this as a type. "No inspection" is however a part of the *decision space* containing inspection decisions. We see that different inspection types may reveal information about the same unit on this system, but this information may vary in accuracy. Instead of considering the different parts of the bridge as sub-systems in the model, the bridge is considered a single-unit system, and inspection types incorporate that the bridge in reality consisting of several sub-systems. Furthermore, four maintenance actions are available for the bridge and not conditional on the inspections.

Courage et al. (2017) models inspection types that yield different information accuracy *and* information on different sub-system's. The system in question is a crack in an offshore wind turbine (alternatively, one can consider the turbine as the system and that it only needs to be described by a crack to optimize its maintenance). The crack is described by the two components "crack width" and "crack depth". Three types of inspection can be carried out. Two of them provide information about the crack depth, one being cheaper and providing less accurate information than the other. The last type provides information about the crack width. Thus, we see that inspection types may differ in both yielded accuracy, which part of the system structure it considers, and the cost of inspecting.

3.3.3 Periodic inspection optimization

In Chapter 2 we introduce *Periodic inspection* as a decision structure commonly modelled in maintenance optimization. Such a structure requires an even, pre-determined interval for all inspections over a planning period. Formally, let $|\mathcal{N}|$ denote the total number of periods for a maintenance optimization problem and n denote a specific period. Deciding periodic inspection then means setting an interval z, where the system is inspected in every stage that $n \mod z = 0$ and $n \leq \mathcal{N}$. Thus, inspection is performed periodically, at pre-determined, even intervals.

The problem of finding an optimal inspection interval is formulated for both multi- and single-unit systems. F. Wu et al. (2015) optimize the inspection interval of a single-unit system. They consider a pre-defined state-threshold for when the system should be maintained. Imperfect maintenance actions are a possibility, making the deterioration process slower without improving the system's state. Yang et al. (2018) optimize several inspection intervals and thus relax the traditional condition of periodic inspection. They allow two different intervals where the shorter interval is applied whenever an inspection (that is first carried out at a longer interval) reveals a single-unit system's defective state.

Considering an infusion pump at a hospital as a multi-unit system, Taghipour and Banjevic (2011) find the optimal inspection interval that enables one of three maintenance actions. In their model, inspections are the only way to gain information about the system, as failures are hidden. Babishin and Taghipour (2016) adopt a combination of both hidden and self-announcing failures by modelling a structural dependency in their multi-unit system. The system works as long as a sufficient number of its components do. If too many components fail, so does the system. System failure is self-announcing, while component failures are not as long as the total number of component failures do not exceed a specific limit. The model optimizes two types of decisions simultaneously; choosing between replacing and repairing a failed component and setting the optimal interval for inspections. Golmakani and Moakedi (2012) also optimize periodic inspection but consider a multi-unit system with stochastic dependency.

The problem of maintaining offshore wind turbines described by Courage et al. (2017), is also a multi-unit maintenance optimization problem that optimizes periodic inspection. Here, an optimal rule is derived, stating which maintenance action that should follow different inspection outcomes. This rule results in three different optimal decisions. The decisions are dependent on the information returned by an inspection (the crack depth). Three different intervals of an increasingly deeper crack detected by inspection result in deciding either no, a minor or a major repair of the crack.

3.3.4 Sequential inspection optimization

In Chapter 2, we also introduce *Sequential inspection*. Such a structure allows the decision of the next inspection at the time of an inspection, independently of earlier inspection decisions. Formally, each stage n can be assigned an inspection variable y_n that can represent each of the possible inspection types by discrete integers. y_n may be a binary variable where 1 means "inspect in stage n" and 0 means "do not inspect in stage n".

Research on maintenance optimization problems that consider sequential inspection optimization is generally more recent than periodic inspection optimization. The single-unit system model by H. Ellis et al. (1995) allows inspection every other year. Although the decision is periodical, the problem considers sequential inspection. Inspections do not have to happen every other year, and thus the inspection intervals are allowed to vary. The model finds optimal inspection and maintenance decisions for every year on this single-unit system.

Berenguer et al. (1997) also consider a single-unit system and use a numerical method to optimize two choices: The date of the next inspection, and whether to replace the system after an inspection. The model assumes self-announcing failures, perfect information and an infinite time horizon. The latter assumption differs from the preceding example and is seemingly not particularly focused on in the literature. This may be because it generally is harder to optimize sequential inspection for a finite time horizon than for an infinite one (Nakagawa and Mizutani, 2009).

It is possible to formulate a dynamic program that considers inspections as a decision like the model from Bloom et al. (2006). Their paper considers inspection as a sequential option not required in order to maintain the underground cable unit. The dynamic program returns an optimal policy where decisions are dependant on the system's age and prior failures. If an inspection (termed "test", in the paper) is conducted, maintenance is immediately carried out if the deterioration is severe enough.

Although several models exist for sequential inspection optimization, we see that most of the maintenance optimization models consider periodic inspection. This is especially the case for multi-unit systems. Durango-Cohen and Madanat (2008) show that adopting sequential inspection can save costs. In their case, relaxing the constraint that requires periodic inspection leads to a reduction in expected cost from the single-unit model's dynamic program.

de Jonge and Scarf (2020) suggest that future research should consider optimal dynamic inspection schedules (instead of the more common heuristic approach). They only review maintenance optimization models with sequential inspection for single-unit systems. This indicates that such optimization for multi-unit systems should be researched further.

3.4 Markov decision process

Using a Markov Process to model transitions between discrete states is a common modelling approach for maintenance optimization. This section describes the approach and shows how to use it when modelling decision-making for a stochastically deteriorating system. Section 3.4.1 explains the characteristics of a decision process, while Section 3.4.2 introduces the Markov process. Section 3.4.3 goes through *how* one can model decision making before an illustrative example is given in Section 3.4.4. Section 3.4.5 gives an overview of maintenance optimization literature that models a Markov Decision Process.

3.4.1 Decision process

With stochastic deterioration as described for systems subject to maintenance, their state or performance may worsen with time. When evolving in a probabilistic manner, a system follows a *stochastic process* (Hillier and Lieberman, 2012). The problem related to making decisions on such a system can be modelled as a *decision process*. Decision processes can describe the system's condition and how it changes over time with the decisions. The system's condition is commonly described by the term *state*. We consider a car tire as an example. Its state can be characterized as either "good as new" or "broken".

Depending on how one chooses to model the state space, one can represent the system's state by a continuous interval of everything between the two characteristics, or only a discrete space where a system is in either state. Models commonly contain something between these two extremes, with several additional discrete steps representing a gradual transition. For instance, we may represent the car tire's condition by two additional states between "good as new" or "broken", these being "worn" and "well worn".

Discrete time-steps such as hours, days or decades usually represent the system's development when making decisions and revealing new information. The total time in question for a system and the possible decisions along the way is called the *planning horizon*, and the time steps are referred to as *stages* (Mes and Rivera, 2017). King and Wallace (2012) point out that *time periods* should not be confused with *stages*, where the former only model time passing by, and the latter model points in time where decisions are made based on new discoveries. For the car tire example, it could be natural to consider seasons as stages, and suggestively eight years as the planning horizon (U.S. Department of Transportation's Federal Highway Administration, 2018; Leister, 2018).

At each stage, a system's development will depend on earlier actions on the system. Depending on the process, a model of this development from one stage to another may include a random variable. Considering a system at a particular stage, the system's further development is affected by the actions made at this specific stage. A given set of actions limits the possibilities of influencing the system's development. This set is known as the *decision space*, where each action corresponds to a *decision*. The agent seeking to optimize the decisions according to some specified objective is called a *decision maker*. If the system model in question has more than two stages where decisions are allowed, it follows a *multi-stage* decision process. Processes where the decision maker only interacts with the system once or twice are single- and two-stage processes, respectively.

3.4.2 Markov process

If a system deteriorates stochastically, its current state is dependant on both earlier actions and the random events affecting its deterioration. At a certain stage, the system's state in the next stage depends on the decisions made in the current stage and the random deterioration that will happen before the next stage. Taking all these factors affecting the system's state into account, modelling the system's state in the next stage means considering many known decisions and random events. To simplify a model describing the system's development, one can assume that the state in the next stage only depends on the current state and the decision made at the current stage. That is, the stochastic deterioration is *not* affected by how the system has come to be in its current state.

With the probabilities of further deterioration only being dependant on information from the current time stage and decisions taken at this stage, the system's development is also only dependant on the current time stage's information and decisions. A system that has this property has a *memoryless property*. The systems then follows a *Markov process* (Hillier and Lieberman, 2012). The term "memoryless" stems from the fact that *how* the system has reached its current state is irrelevant and can be "forgotten". Using the car tire as an example, its probability of further deterioration is not dependent on whether it was "as good as new" in year two or if it was "worn" in year two when we see that is it "well worn" in year three.

We can formally consider the random variable X and a subset $S \subseteq [0, \infty]$. In the case of deteriorating systems, X may represent the state with higher values meaning further deterioration. S may be the possible outcome of events affecting the system, and the sum of two outcomes means that both have happened. According to Nelsen (1987), if the following equation holds:

$$P(X > a + b | X > a) = P(X > b), a, b \in S, \tag{3.1}$$

the variable has the *Markovian property*. Using the car tire as an example with X representing the state of the tire, a may be that the tire at some point has been "good as new" and b may be that now it is "well worn". Furthermore, we assume that the tire may deteriorate directly to any state representing a worse condition than its current condition. The probability of the tire being "broken" (that is, worse than b), is the same regardless of the fact that the tire may at some point have been "worn" or gone directly from "good as new" to "well worn". In the illustrative case of the car tire, Equation 3.1 only holds if the most recent observation of the car tire yielded "well worn" (that is, $X > a$). This is because of the limited state space and the fact that the current stage affects the deterioration probability.

It is important to note that past states are taken into account by the present state. While we do not consider past states when calculating probabilities for future states, they do have indirect importance as the current state results from past states and state transitions.

3.4.3 Making decisions

The state space considered for a specific problem may be finite as suggested for the car tire, or infinite such as could be the case when using the age of a system to represent its state directly. Assuming that the state space is countable or consisting of a finite set of states and that the modelled deterioration process is a Markov process, the process is called a *Markov chain* (Petrushin, 2000).

Markov chains may be either *discrete time* or *continuous time*, depending on how the observations of the system's state are made. A system may be subject to continuous monitoring or inspected at a pre-determined interval (such as the suggested seasonal inspection of a car tire, possibly in conjunction with exchanging winter and summer tires). When using a Markov chain to model a decision process, decisions can be pre-determined given different observations at different stages by optimizing for the objective, taking all future possibilities into account. A decision process that uses *transition probabilities* to calculate how *immediate* and *subsequent* effects will contribute to the objective function, is called a *Markov decision process* (MDP).

In a stochastic process, the system state in the next stage will be dependant on a random event. Any decision will also affect the model's objective function, commonly as a cost or a reward. Some decisions may be costly, but lower the probability of events expected to exceed these decisions' cost. Decisions influence the probability of future states and consequently, the expected cost of future decisions. An MDP seeks to find the decision sequence that optimizes the objective, which often is to minimize expected costs, but may also be another pre-defined criterion. Any sequence of decisions is called a *policy*, a term introduced by Bellman (1954). A sequence of decisions that optimizes the objective is an *optimal policy*.

With the above-described characteristic, we can formulate the problem of finding the optimal policy mathematically. The problem considers making the optimal decisions on a system within a planning horizon, given that it develops through a stochastic process with discrete states and transition. In a stage in the planning horizon, $n \in \mathcal{N}$, the system will be in one state s_n within the state space $\mathcal{S}$. A decision's cost is given as $C_n(S_n, x_n)$. A valid decision, x_n, is limited to decisions in the decision space, $\mathcal{X}_n$. The following state in our MDP depends on a probability distribution for going to other states $\mathbb{P}(s_{n+1}|s_n, x_n)$. The combination of a random event and the decision made in the current stage, brings the system into a state, s_{n+1} in the next stage $n+1$. Note that the state in the next stage may be the same as the current state. The decision function $X^{\pi}(s_n)$ returns a decision for all states. $\pi \in \Pi$ denotes a sequence of decisions in the set of all possible sequences. Thus, π denotes a policy and Π the set of all possible policies. Finding the optimal policy is then equivalent to solving the expression (Mes and Rivera, 2017):

$$\min_{\pi \in \Pi} \mathbb{E}^{\pi}\{\sum_{n \in \mathcal{N}} \gamma^n C_n(s_n, X_n^{\pi}(s_n))\} \tag{3.2}$$

In Equation 3.2, the factor $\gamma \leq 1$ can be interpreted as a *discount factor*, allowing future rewards to be discounted reducing the impact of costs with time. Omitting γ (that is, implicitly setting $\gamma = 1$), the derived expression will treat costs as equally contributing to the objective over the planning horizon, and consequently minimize the total expected costs. Altering Equation 3.2 to consider an infinite planning horizon (straightforwardly obtained by setting $\gamma \leq 1$), the expression will minimize the long-run average cost.

3.4.4 An illustrative example of a Markov decision process

To obtain a simple illustration of a Markov decision process, we consider a system that only consists of one component. Throughout a year, the system will either continue to work normally or fail. In the case of a failure, one must replace the system with a new system. For simplicity, we assume that in the case of a failure, the system will not fail again before the next year and still be possible to maintain in the same period (although it will likely be left alone, as the age will be very low at the beginning of the next year). The probability of failing is a known function that solely depends on the age of the system. At the end of every year, the decision maker must choose between replacing the system, or leaving it in its current state. Considering the system's age as its state, the possible state transitions are ageing or replacement (where the latter implies a brand new system at the beginning of the next year). This also means that the transition probabilities at a given point of time for the system, considering a year forward in time, are given from only the decisions taken at this point and the system's age.

Given the characteristics described above, we have a Markov decision process. Technically, both the state space and planning horizon are infinite; we have not defined any end of the planning horizon, or a maximum age allowed system age. Let each stage be a year, represented by n. The state of the system, s_n, is defined by its age. Costs either incur when replacing the system, or a failure occurs. Note that in order for the model to yield any other policy than simply doing nothing each year, the cost of failure must exceed the cost of replacing the system. In a stage n, the system will be in state s_n, and we represent decision in this state by x_n. As the system's future states are only dependant on the current state and decision, so is the expected future cost. This cost can be formulated as $c_n(s_n, x_n)$. Furthermore, we can represent the transition probabilities with the two expressions $P^F(s_n)$ and $1 - P^F(s_n)$ for a system failure and normal ageing, respectively.

To illustrate the stages of the proposed MDP, we use a *decision tree*. Figure 3.1 shows a decision tree for three stages in the planning horizon, corresponding to the system described above. The squares are *decision nodes*, denoting times when the decision maker can make a choice between replacing the system ($x_n = 1$ for stage n) and doing nothing ($x_n = 0$ for stage n). The circles are *chance nodes*, illustrating that a probabilistic event decides the immediately succeeding state of the system. Arcs between the nodes illustrate the possible state transitions for a system in a given state to another. The probability of moving along an arc going out of a choice node lies above the appurtenant arc.

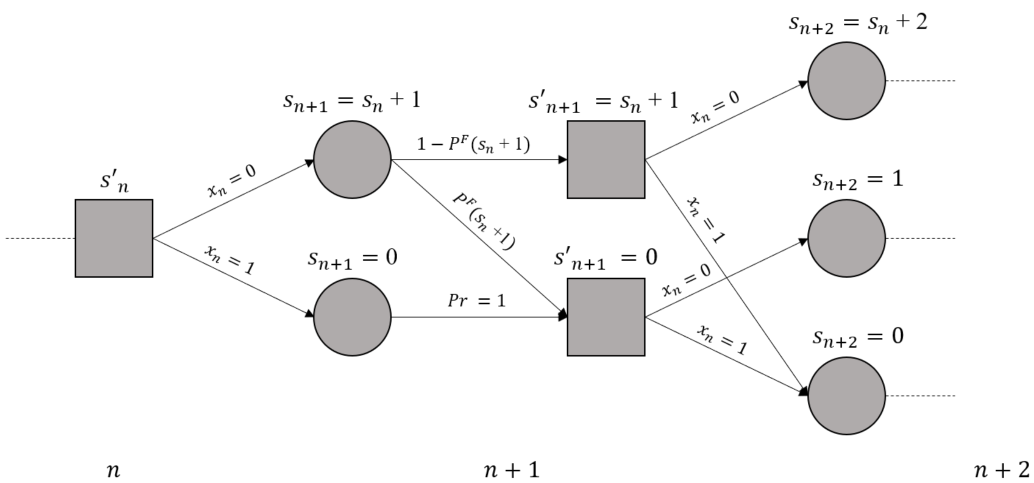


Figure 3.1: Markov decision process illustrated as a decision tree

To highlight the difference between decisions and random events, we mark some states, s'_n. This marking distinguishes between the course of events within a stage. As mentioned, a failure may happen throughout a year, but the maintenance decision is made at the end of the year, meaning with the system in state s'_n in year n. This order of events makes it possible to maintain the system in the same year that a failure forces its replacement. Although this is a result of how we model the process, considering what happens within a single stage is vital to obtain a model that corresponds as closely as possible with the real-life situation.

3.4.5 Markov decision process in maintenance optimization literature

In the literature survey by Pierskalla and Voelker (1976), maintenance optimization models are classified based on how they model deterioration. One of the modelling approaches addressed in this particular classification is the use of an MDP. This use implies that the Markov Decision Process is a traditional modelling approach for maintenance optimization processes. However, the same approach is commonly used in newer publications, possibly incorporating uncertainty in observing system states in a stage.

A system is said to be *Markov deteriorating* if the deterioration process is Markovian (Bloch-Mercier, 2002). Such a modelling approach is found in Chiang and Yuan (2001) and Kurt and Kharoufeh (2010). The latter publication considers a system subject to periodic inspection and Markov deterioration. The system's state space is discrete, and a state represented by a high numerical value indicates a "worn" system with an increased probability of failing. A system in the state with the largest possible value has failed and does not function. The objective is to derive a replacement policy that minimizes the total discounted cost over an infinite horizon, regarding costs of inspections, replacements and operation.

Maillart (2006) considers an infinite planning horizon in his model. It describes the maintenance of a system as a Markov chain with discrete time steps. Inspections reveal the system's state and imminently require the decision maker to either maintain the system or do nothing. The paper regards inspection as a choice as well, and discuss two cases. First, the proposed model considers information revealed through inspection as perfect. Later, an extension of the model considers a probability distribution for states the system may be in. An MDP is formulated and minimizes cost for the infinite horizon.

We see that even stochastic processes with uncertain information reveal may be MDP-models. If the state distribution is based on uncertain observations of the system's state, the process is "hidden" (Eddy, 2004). When formulating such a Markov Decision Process, the corresponding stochastic model that describes state transitions from one stage to another is a *hidden Markov model* (HMM) (Petrushin, 2000). An MDP where a systems develops according to a HMM is a *partially observable Markov decision process* (POMDP) (Sondik, 1971). Several recent publications consider this approach (Byon and Ding, 2010; Andriotis and Papakonstantinou, 2020; Morato et al., 2020). The uncertain observations may be something else than inspection, e.g. self-announcing failures. Using Hidden Markov Model theory, Neves et al. (2011) estimates a single-unit system's state by using empirical data.

3.5 Dynamic programming

This section considers a common solution approach to maintenance optimization problems applicable to MDPs. We first introduce the theory behind the approach in Section 3.5.1 before presenting how a dynamic program can consider uncertainty in Section 3.5.2. Section 3.5.3 provides an example illustrating how to find optimal decisions in an MDP by using stochastic dynamic programming. We discuss a commonly discussed drawback of dynamic programming called "the curse of dimensionality" is discussed in Section 3.5.4 before the section is concluded by providing publications that use dynamic programming to solve maintenance optimization problems in Section 3.5.5. We present publications considering single-unit systems before turning the attention to multi-unit systems.

3.5.1 Introduction to dynamic programming

One common approach to solve an MDP to optimality is using *dynamic programming*. The overview of the literature on maintenance optimization suggests that modelling a maintenance decision problem as an MDP and using a dynamic programming algorithm is suitable for such problems. This approach is, among others, suggested by Mes and Rivera (2017). Dynamic programming is a method developed by Richard Bellmann in the 1950s, motivated by studies of multi-stage decision processes and the relating mathematical challenge of solving problems related to these processes (Bellman, 1954). Bellman used a repetitive technique to derive a *functional equation* for dynamic programming. He decided to call the technique "The Principle of Optimality" (Dreyfus, 2002). Another name for this principle is "The Bellman Principle":

"*An optimal policy has the property that whatever the initial state and initial decisions are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decisions.*" (Bellman, 1954, p. 504)

One common approach to solve an MDP to optimality is using *dynamic programming*. The overview of the literature on maintenance optimization suggests that modelling a maintenance decision problem as an MDP and using a dynamic programming algorithm is suitable for such problems. This approach is, among others, suggested by Mes and Rivera (2017). Dynamic programming is a method developed by Richard Bellmann in the 1950s, motivated by studies of multi-stage decision processes and the relating mathematical challenge of solving problems related to these processes (Bellman, 1954). Bellman used a repetitive technique to derive a *functional equation* for dynamic programming. He decided to call the technique "The Principle of Optimality" (Dreyfus, 2002). Another name for this principle is "The Bellman Principle":

$$f_n(s_n) = \min_{x_n \in \mathcal{X}_n} \{f_{n+1}(T_x(s_n))\} \tag{3.3}$$

As all possible valid transformation are taken into account, Equation 3.3 gives a deterministic dynamic program (Bellman, 1954). In the MDPs discussed so far, we see that transitions are dependant on decisions. To more explicitly express a relation between decisions and transformations, one may use a *transition function*, $s_{n+1}(s_n, x_n)$ (Lundgren et al., 2012, p. 485). Given state s_n and decision x_n, the function describes the resulting state s_{n+1}. Representing the immediate cost of choosing x_n in state s_n at stage n by $c(s_n, x_n)$, the recursive relation can be formulated as (Hillier and Lieberman, 2012, p. 438-468):

$$f_n(s_n) = \min_{x_n \in \mathcal{X}_n} \{c(s_n, x_n) + f_{n+1}(s_{n+1})\} \tag{3.4}$$

From Equation 3.4, we see that the return from being in state s_n at stage n is dependent on the return obtained in the next stage, $n+1$. Thus, the solution procedure is required to start at "the end" and find an optimal policy at the last stage in the planning horizon, $|\mathcal{N}|$ (which will only contain one decision and transformation). After this policy is derived, it is used to derive the policy for stage $|\mathcal{N}|-1$. To obtain the optimal policy for the whole planning horizon, polices must be derived stepwise for earlier stages until the optimal policy for the *initial* stage is returned. Because the solution approach moves recursively backwards in time, it uses *backward recursion*. As the optimal policy at stage $|\mathcal{N}|$ also is "forward-looking" according to Equation 3.4, it implicitly requires a final value to initialize the recursion. This value, $f_{n+1}(s_{n+1})$ for $n = \mathcal{N}$, may be set for each different possible state in the state space.

Originally, Bellman (1954) formulated the functional equation by using *forward recursion*. This is an alternative formulation of Equation 3.4. Instead of a final value $f_{n+1}(s_{n+1})$ for $n = \mathcal{N}$, a formulation using forward recursion requires an initial value given for $f_0(s_0)$ (assuming that the first decision stage is at $n = 1$). The forward recursion approach moves chronologically along the stages until the last stage.

3.5.2 Stochastic dynamic programming and relation to MDP

State transitions in an MDP may either only be dependant on the decisions taken in a stage, or additionally a random event in each stage. In the latter case, both a decision variable and a random variable drawn from a *probability distribution* will influence state transitions from one stage to the next. When decisions alone do *not* determine these state transitions, one may still use dynamic programming as the solution approach. The program is then a *stochastic dynamic program* (SDP). In the literature, this is also termed a *probabilistic dynamic program* (Hillier and Lieberman, 2012, p. 438-468). This kind of dynamic program was suggested by Bellman (1954) as an early extension to the original formulation. Instead of relying on certain values, the return function, $f_n(s_n)$ yield the *expected value* of being in state s_n in stage n.

With the adoption of $f_n(s_n)$ representing expected cost, Equation 3.4 becomes a valid solution to the expression derived for an optimal policy in an MDP (see Equation 3.2 in Section 3.4.3). By recursively finding optimal decisions in all stages, starting with the last stage, Equation 3.4 only holds if the decisions x_n for $n \in \mathcal{N}$ together make up an optimal policy, corresponding to to X_n^{π} in Equation 3.2. As the cost function $C_n(s_n, x_n)$ yield expected cost, it can be represented by the adopted $f_n(s_n)$. By dividing the problem of finding an optimal policy for the MDP into smaller *sub-problems*, it is solved through stochastic dynamic programming.

As optimal policies are derived recursively from the end to the beginning, the program will also yield optimal policies for all the sub-problems (Mes and Rivera, 2017). Thus, after solving the original problem, a dynamic program has derived an optimal policy for *any* valid state for every stage subsequent to the first stage considered in the original problem. All these policies may be beneficial if one is to make informed decisions in every stage. As the decision maker can adapt to improbable events assigned a low expected cost (because of the low probability of the event happening), the fact that the SDP calculates *all* optimal policies can be exploited. However, this also means that the recursive function must calculate the expected cost of all decisions for all possible events, no matter how unlikely they may be. This naturally results in many calculations, a challenge further addressed in Section 3.5.4.

3.5.3 Using dynamic programming to find optimal decisions in an MDP

Considering the single-unit system example introduced in Section 3.4 to illustrate an MDP, we now turn our attention to how to solve it with dynamic programming. The decision maker faces the choice of replacing the system every year. For a new system, a system of age 0, the chance of failure is 0. When the system reaches 1 and 2 years of age, the respective chance of failure before reaching the next stage is 0.4 and 0.6. We assign a cost of 30 for replacing a failed system and 15 for replacing a functioning system. Although a both of these costs represent a replacement, the higher cost of replacing a failed system may be explained simply by the cost of not knowing *when* in a year a failure occurs, or a penalty cost that incurs when the system is not delivering its desired output.

Adopting the notation from the recursive relation in Equation 3.4, $f_n(s_n)$ is the expected cost in stage n of making present and future decisions according to the optimal policy. Let $c(s_n, x_n)$ be the expected cost of decision x_n in stage n. Equation 3.4 then gives the optimal policy for maintaining the single-unit system.

The problem must have an *initial condition* to initialize the recursive procedure. Assuming that we consider only decisions on the system in year 1 and 2, our planning horizon $\mathcal{N}$ is limited to the set $\{1,2\}$. We denote the initial condition as $f_3(s_3) = c_{s_3}$, assigning a cost of being in state 0, 1 and 2 at the end of the second year. Figure 3.2 presents the initial conditions and the solution approach. At stage 3, we calculate the expected cost of being either of the possible states at the beginning of stage 3 and then repeat the process for stage 2 and stage 1.

In the figure, moving along an arc (as either a result of decision or chance) has an associated cost. The cost for an arc is placed above it. Arcs going out of a chance node are also assigned a probability corresponding to the transition probabilities but omitted in the figure for illustrative purposes. At least one arc going out of a decision node is red, denoting the optimal choice. We see that if the system fails the first year (or one makes the non-optimal choice of maintaining the system at the first opportunity), the expected cost of both choices in year 2 is 20, and thus any choice is optimal. Furthermore, achieving the lowest expected means letting the system age once and then replace it. The optimal policy comprises these two choices.

In year 0, the expected cost of following the optimal policy is 32. This expected cost includes the expected cost of optimal choices in all possible subsequent stages. To illustrate the expected cost calculation, we consider the expected cost of being in the worst possible state in the last stage (the rightmost upper circle). The probability of failure is 0.6, and the cost of failure is 30. The initial conditions state that failure and repair will mean a cost of 5 being incurred at the end, while the cost of staying in the worst state means a cost of 15. As the probability of no system failure is 0.4, the expected cost is: $0.6 \cdot (30+5) + 0.4 \cdot 15 = 27$.

It is important to note that the optimal policy does not guarantee the least costly outcome. If a failure occurs in the first year when following the optimal policy, the minimum expected future cost is 20, but the penalty cost of 30 will occur due to system failure. Thus, in this situation, the expected total cost will be 50.

We see that deciding to replace the system each year would cost 35 with certainty, providing a better outcome than following an optimal policy when a failure happens in year 2. This observation illustrates the difference between an optimal solution and a guarantee of obtaining the lowest possible cost. Still, when faced with the first decision, the expected cost was 32, better than the 35 that proved to be the lowest possible cost when following an optimal policy. It is also interesting to observe that when left in an unlikely state (that is, the less likely event has occurred) in stage 2, the decision maker can still follow an optimal policy for minimizing future costs.

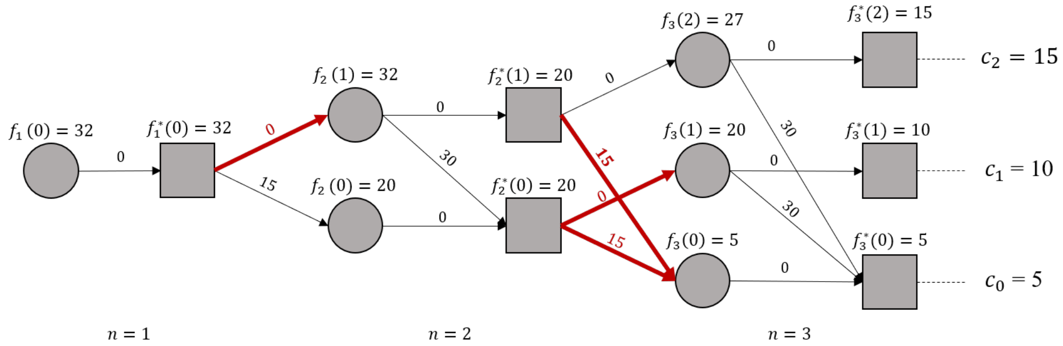


Figure 3.2: The figure shows how to use dynamic programming to minimize the expected costs in an MDP. Red vectors indicate optimal decisions at each stage.

3.5.4 The curse of dimensionality

While calculating the optimal policies for all possible combination of events can be beneficial, the complexity of doing so is a widely discussed drawback of dynamic programming. As stochastic dynamic programming considers probabilistic events in addition to the possible decisions, this solution approach becomes especially complex for large problems. The need to calculate expected costs for all combination of events is known as the *curse of dimensionality*. Powell (2016) divides this curse into three separate curses, being:

- **The state space.** Consider a multi-unit system consisting of $|\mathcal{K}|$ components, where each component has $|\mathcal{S}|$ individual states. The state space representing the whole system has the size of $|\mathcal{S}|^{|\mathcal{K}|}$ different possibilities.
- **The outcome space** A specific system state may be associated with several transition probabilities, representing the possible states that the system may transition into in the next stage. These possibilities for all state transitions make up *the outcome space*, and a dynamic program must calculate the expected return for all outcomes in each stage.
- **The decision space** The illustrative examples introduced so far have only considered two possible actions; replace or do nothing. However, a model can consider several possible maintenance decisions. An example is adding a "minimal repair" option, as considered by Sheu et al. (2018). Replacements and minimal repairs may differently affect the probability distribution for future outcomes. Thus, each possible decision may differently affect the expected return of

subsequent stages. As is the case for the outcome space, the dynamic program must calculate all of these expected returns.

3.5.5 Dynamic programming in maintenance optimization literature

Considering a single-unit system, Chu et al. (1998) formulate a dynamic program approach for minimizing the cost of maintenance, where the possible maintenance actions are to repair a failed system or carry out preventive maintenance. The paper concludes that considering several maintenance possibilities with different costs would be the next step, studying a system consisting of two components. In another single-unit formulation, (Bloom et al., 2006) consider an underground distribution cable subject to several decisions: Inspection, repair, rejuvenation or replacement. A dynamic program encapsulates both forecasts of deterioration and information reveal as a result of inspections. The program minimizes the life-cycle cost for a cable. More recently, the research on maintenance optimization models of single-unit systems uses stochastic dynamic programming to investigate problems with finite time horizons (de Jonge and Scarf, 2020). Stochastic dynamic programming may exploit uncertainty of maintenance outcomes. That is, a preventive maintenance action can only detect *some* deficiencies and thus may be unsuccessful. This can result in a system failure despite recent preventive maintenance. Sachan et al. (2016) suggest an SDP for this particular problem.

Facing the decision of long-term scheduling of maintenance, Janjic and Popovic (2007) argue that dynamic programming is expedient for their multi-unit system with discrete states and consequently discrete transitions. Encapsulation both of the suggested next steps from Chu et al. (1998), Korpijärvi and Kortelainen (2009) consider a multi-unit system and allow for both repairs and replacements. They minimize cost by the use of dynamic programming. As dynamic programs are often applied to maintenance optimization problems, the curse of dimensionality has also received attention in related literature. To curtail the complexity of large models, Abbasi et al. (2009) suggest a program for a multi-unit system that exploits risk management theory to prioritize different maintenance decisions.

Chapter 4

Problem Description

This chapter presents the Utility Mast Inspection and Maintenance Problem (UMIMP). In Section 4.1, we present the characteristics of the UMIMP. In the next section, we describe the problem's decision structure.

4.1 Problem characteristics

The UMIMP is to find the optimal inspection and maintenance decisions for a power grid utility mast. The objective is to minimize the total expected costs from failure, inspection and maintenance over a *planning horizon* while also accounting for the utility mast condition at the end of the planning horizon.

A power grid line consists of several homogeneous utility masts, with a set of *components*. We consider a collection of components representing the mast a *system*. Some of the mast's components affect its ability to deliver electricity. If the failure of a component results in the mast's failure and, consequently, the power grid line, it is considered critical. The problem considers critical components, as they have the most significant implications for inspection and maintenance decisions.

A component is always in a certain condition. A components' condition is subject to change over time due to stress and damage from external factors. That is, it *deteriorates*. A component in a significantly deteriorated condition will be more prone to failure than a less deteriorated component. The most important external factors are weather, wildlife and natural deterioration processes. A components' exposure to such factors is stochastic, hence so it is deterioration process. For example, a woodpecker might peck the mast's wooden pole, which worsens the poles' state. Before a certain point in time, we cannot tell if the woodpecker will choose to do so on a given pole in the future. However, based on empirical data, we can say something about the probability that a component will deteriorate due to woodpeckers or other external factors, during a given period of time.

A set of discrete *states* can model condition of a component. The lowest state indicates the best condition, being a new, or good as new, component. Higher states indicate progressively worse conditions, the highest state being a component that has failed. During a time period, a components state may remain unchanged or change to a worse state. This deterioration is modelled by *transition probabilities* for each component, representing the probabilities of remaining unchanged or worsening to each of the possible worse states during a time period. The probability of deterioration depends on a component's current state, not how long it has been in that state or its previous state development. That is, the transition probabilities have the Markovian property.

When a power grid line fails to deliver power, its operator pays a penalty cost of energy not supplied, *CENS*, based on the number and importance of the affected customers and increases with the grid's total downtime. A power grid operator seeks to maintain the mast components so that the costs spent on inspection and maintenance are less than the reduction in expected cost from CENS and unplanned maintenance.

Deterioration mostly affects the risk of failure and not how the components function. A pole can be quite rotten and still completes its function (holding the grid lines and other components above the ground). However, it is much more likely to break when exposed to external forces like wind than a new pole. Additionally, sudden events like a lightning bolt might break even a new mast. A new mast has a lower risk of failure than an older, more deteriorated mast because no deterioration has happened. However, the risk of a new mast failing is still present.

4.2 Decision structures

The problem concerns three types of interactions with the power grid: *repair* of a failed system, *maintenance* to reduce failure risk and *inspection* to uncover the state of the mast's components. If a system failure occurs, a repair is needed to restore the power grid's function. As opposed to maintenance and inspection, a repair is not a choice and must be carried out as fast as possible after a failure occurs.

Maintenance takes place before a component has failed, improving its state. As components in better condition have a lower chance of failing, preventive maintenance reduces the risk of, and consequentially the expected cost from, failure.

Inspections detect the states of the system's components. They have an associated cost, but no direct effect on the power grid. However, an inspection reveals information about all system components and can help a power grid operator decide on maintenance and lower the risk of failure.

Maintenance can only be carried out within a given time period following an inspection. The information that an inspection reveals aids the maintenance decision. The maintenance cost depends on the type of component being maintained and the time spent by the personnel maintaining it. Also, any maintenance has an associated setup cost independent of the number of components maintained. When deciding on maintenance, the operator can choose to maintain any set of components, allowing several components to share the same setup cost.

Inspection and maintenance decisions should be seen in relation to each other. Increasing the frequency of inspections will make a power grid operator more capable of assessing the components' state and thus their risk of failure. If an inspection reveals that a component is in a lower state than expected, the operator can postpone maintenance of the component. This postponement can reduce total maintenance costs. Components in a worse state than expected can be maintained, reducing the risk of failure and expected costs. However, increasing the frequency of inspections also means increased inspection costs.

The power grid operators will typically decide on an inspection plan, inspecting the grid at specific intervals. They also face regulations imposing inspections at least every tenth year. However, they are not obligated to take any action regarding the grid's maintenance after an inspection.

The operators may inspect their grid more often than the regulations require. This option also means that they may decide when the next inspection should be conducted based on revealed information from a preceding inspection. Thus, both the maintenance decision and the next inspection's timing can be adjusted based on the insights from an inspection.

Chapter 5

Mathematical Model

This chapter presents two mathematical models for solving the Utility Mast Inspection and Maintenance Problem (UMIMP). Section 5.1 explains our general modelling approach and Section 5.2 presents assumptions for both models. Section 5.3 presents the Periodic SDP (PSDP), a stochastic dynamic programming model for the problem of optimizing periodic inspection intervals and maintenance decisions. In Section 5.4, we solve two simple problems with the PSDP to provide a better understanding of how it works. Finally, Section 5.5 presents the Sequential SDP (SSDP) that builds on the PSDP but allows inspection intervals of varying length, that are set dynamically depending on the state of the system.

5.1 Modelling approach

The UMIMP involves maintenance and inspection decisions throughout a planning horizon. Hence, the problem is inherently multi-staged. Also, deterioration and failures are stochastic; we only know their respective probabilities. Given these problem characteristics, we propose to use a stochastic dynamic programming approach to solve the UMIMP.

We look at a system of different components throughout a planning horizon. Discrete states model the condition of a component. We divide the planning horizon into equal length periods, stages, in which a selection of events that influence the state of the components may occur. Deterioration and failure events are subject to randomness, while a decision maker's choices determine inspection and maintenance events. In our models, the events in a stage follow a pre-defined order. Figure 5.1 shows the events and their order.

Depending on a given inspection policy, an inspection may take place at the beginning of a stage.

An inspection reveals the system's state, i.e. the pre-maintenance state becomes known. Then, based on the pre-maintenance state's value, a model suggests a maintenance decision for each component. If the model suggests maintaining a component, it is replaced and considered "good as new". If no inspection takes place, the state of the system remains unknown, and no maintenance takes place.

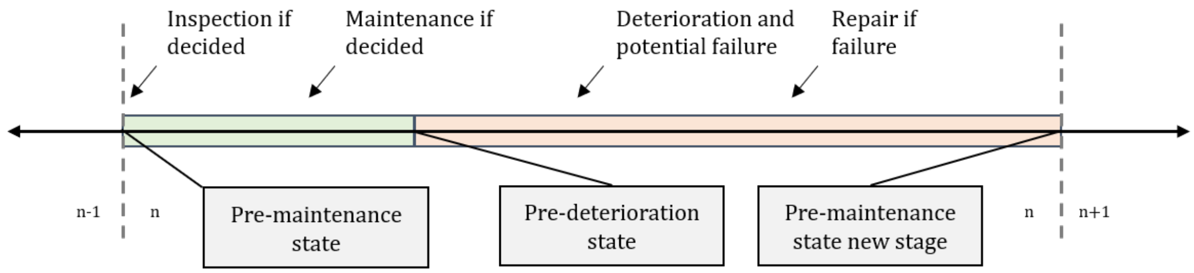


Figure 5.1: Timeline indicating the order of which different events influence the system

The system is now in what we call the "pre-deterioration state". The components may now deteriorate, and in the worst case, fail. The probability of deteriorating differs for different components and depends on the component's current state. If a component fails, it results in a system failure. This failure incurs a penalty cost and forces a system repair. Repair happens directly after failure, making the component "good as new". After deterioration and repair of potential failures, the resulting state is the next stage's pre-maintenance state.

5.2 Modelling assumptions

The following assumptions apply for both the Periodic and Sequential SDPs.

- **Modelled components are critical.** All modelled components are considered critical for the system's ability to function. If a component fails, the system fails as well.
- **Failure is handled individually for each component.** When several components fail within the same stage, they do so at different times. Thus the cost of failure can be incurred several times per stage for different components. The probability of two components failing at the exact same time is negligible, meaning that there are no synergies to be exploited when a failure occurs. Therefore, each failure will incur CENS, the components variable maintenance cost and the fixed maintenance cost.
- **Maintained components become "good as new".** When maintaining a component, it is set back to the first state and is indistinguishable from a new component.
- **Repair of a critical component is carried out immediately.** When the system fails, repair of that component will be done momentarily, incurring associated costs.

- **A component can only fail once per stage.** The probability of a component failing, then being repaired, and then failing again in the same period is negligible.
- **CENS is constant.** The penalty cost of failure is constant for all components in the system, regardless of which component caused the failure. The cost is based on empirical data, but do not consider varying downtime due to factors like longer repair times for certain components.
- **Deterioration follows know probability distributions.** While the outcome of a deterioration process in a stage is unknown, the deterioration follows a know probability distribution. Deterioration has the Markov property, as described in Chapter 3 The modelled deterioration of a component is only dependant on the state of the component and independent of age and deterioration of other modelled components.
- **Time is discrete.** The planning horizon is divided into a finite amount of time periods or stages. These all have the same length and are indexed increasingly, the first period being 1 and the largest index being the final stage.
- **Inspection provide perfect information.** An inspection returns the actual states of all components. As discussed in chapter 3, while this is a simplification, it is a reasonable assumption when modelling a discrete state space.
- **Inspection in first stage.** As the initial state is known, we assume that an inspection takes place in the first stage.

5.3 Periodic SDP model

This section presents a model for solving the UMIMP with periodic inspection intervals by a stochastic dynamic program and is thus referred to as the Periodic Stochastic Dynamic Program (PSDP or Periodic SDP). The model aims to find optimal maintenance and inspection decisions, i.e. decisions that minimize total expected costs over a planning horizon.

5.3.1 Notation

We use the following notation for the PSDP. Most of it is also used for the Sequential SDP presented in Section 5.5, except for explicitly mentioned changes.

Sets:

$\mathcal{N}$	Set of stages $\mathcal{N} = \{1, 2, ..., \lvert\mathcal{N}\rvert\}$
$\mathcal{K}$	Set of components $\mathcal{K} = \{1, 2, ..., \lvert\mathcal{K}\rvert\}$
$\mathcal{S}$	Set of states $\mathcal{S} = \{1, 2, ..., \lvert\mathcal{S}\rvert\}$

Parameters:

C^F	The penalty cost of failure, typically referred to as CENS (cost of energy not supplied)
C_k^{MV}	Variable cost of maintaining component $k \in \mathcal{K}$
C^{MF}	Fixed cost of maintenance
C^I	Cost of inspection
C_{ik}^{End}	End of horizon cost. Cost of ending up in state i for component $k \in \mathcal{K}$
B_k	Initial state of component $k \in \mathcal{K}$
P_{ijk}	Probability that component $k \in \mathcal{K}$ deteriorates from state $i \in \mathcal{S}$ to state $j \in \mathcal{S}$. Summarized in a matrix D_k for each component
z^{max}	The maximum allowed inspection interval

Decision variables:

x_{kn}	$x_{kn} \in \{0, 1\}$. 1 if component $k \in \mathcal{K}$ is maintained in stage $n \in \mathcal{N}$, 0 otherwise. $x_n = [x_{1n}, x_{2n}, ..., x_{\lvert\mathcal{K}\rvert n}]$ is sometimes used for convenience
z	Inspection interval, i.e. number of stages from one inspection to the next. $z \in \{1, ..., z^{max}\}$
y_n	$y_n \in \{0, 1\}$. 1 if inspection takes place in period n and 0 otherwise. Is forced by the value of z, such that between two consecutive inspections there are $z - 1$ stages without inspection

State variables:

s_{kn}	Pre-maintenance state. $s_{kn} \in \{1, ..., \lvert\mathcal{S}\rvert\}$. The condition of component $k \in \mathcal{K}$ at the beginning of stage $n \in \mathcal{N}$. 1 is considered good as new, $\lvert\mathcal{S}\rvert - 1$ significantly deteriorated and $\lvert\mathcal{S}\rvert$ failed. $s_n = [s_{1n}, s_{2n}, ..., s_{\lvert\mathcal{K}\rvert n}]$ is sometimes used as notation for the system's state
s'_{kn}	Pre-deterioration state. $s'_{kn} \in \{1, ..., \lvert\mathcal{S}\rvert\}$. The condition of component $k \in \mathcal{K}$ in stage $n \in \mathcal{N}$ after potential maintenance and before potential deterioration and repair of failures. $s'_{kn} = [s_{1n}, s_{2n}, ..., s_{\lvert\mathcal{K}\rvert n}]$ is sometimes used as notation for the system's state

5.3.2 Model formulation

State transition

Figure 5.2 indicates the order of events that influence the system. In the figure, we use squares to indicate decision nodes, i.e. when a decision maker determines the outcome, and circles to indicate chance nodes, where the outcome is stochastic. If an inspection takes place, as in Figure 5.2 a), it reveals the system's state. The decision maker will then choose which components to maintain, if any, leading to the pre-deterioration state. As the pre-maintenance state is known, and only deterministic maintenance decisions influence the system between s_n and $s_n^{'}$, the value of the pre-deterioration state is also known. Then the system experience deterioration, where each component may or may not deteriorate to a worse state, and potentially fail. If a failure occurs, the component is immediately repaired. The resulting state is the pre-maintenance state of the next stage ($s_{(n+1)}$). Unless an inspection takes place in stage $n+1$, the state of the system is now unknown. Figure 5.2 b) indicates the same order, but in this case, no inspection takes place. Consequently, the state of the system is unknown, and no maintenance decision takes place.

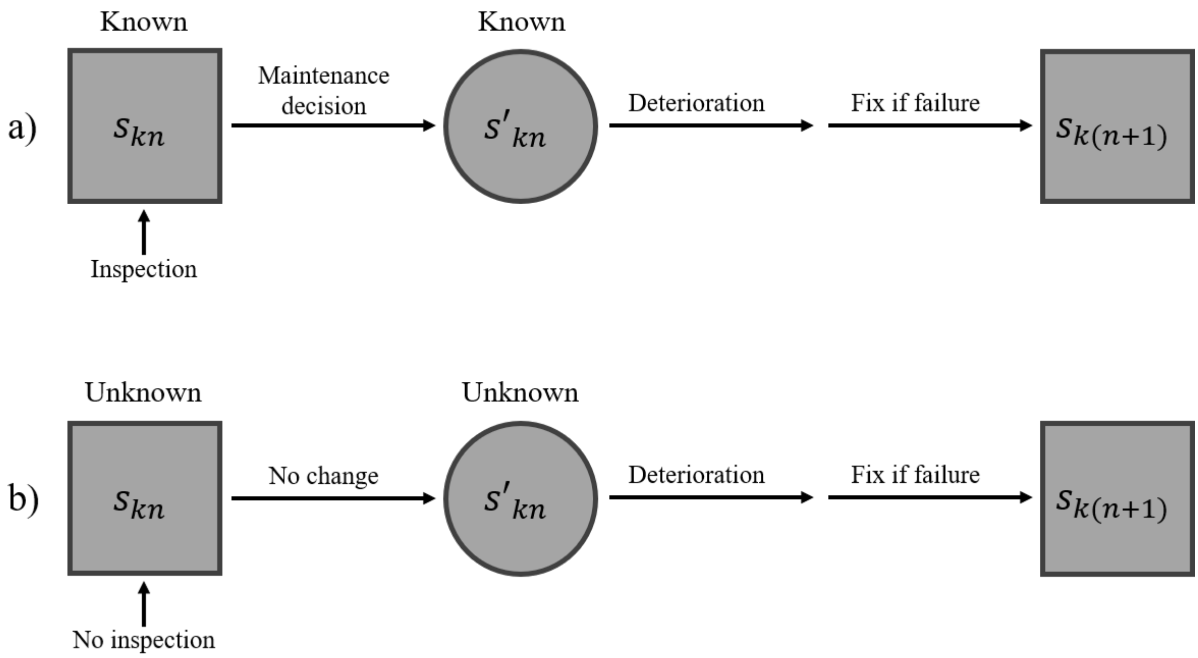


Figure 5.2: The figure shows transition between states in a stage. An inspection takes place in a), but not in b). Squares indicate decision nodes and circles indicate chance nodes.

Function 5.1 describes the transition between the pre-maintenance state, (s_n), and the pre-deterioration state, ($s_n^{'}$), as a result of this decision. If a component undergoes maintenance, it is back to the first state. If no maintenance takes place, the state remains the same.

$$s_{kn}^{'} = \begin{cases} 1 & \text{if } x_{kn} = 1 \\ s_{kn} & \text{otherwise} \end{cases}, \; k \in \mathcal{K}, \; n \in \mathcal{N} \tag{5.1}$$

The state transition from a maintenance is illustrated in Figure 5.3. Considering a component, $k \in \mathcal{K}$, in a stage, $n \in \mathcal{N}$, represented by five discrete states ($|\mathcal{S}| = 5$), Figure 5.3 a) indicates the possible transitions from $s_{(n)} = 1$, b) from $s_{(n)} = 2$ and so forth. When the state is 1, as in Figure 5.3 a), we see that the system will stay in state 1 independently of the maintenance choice. However, for Figure 5.3 b), c) and d), the maintenance choice and its implications for the possible state transitions become apparent.

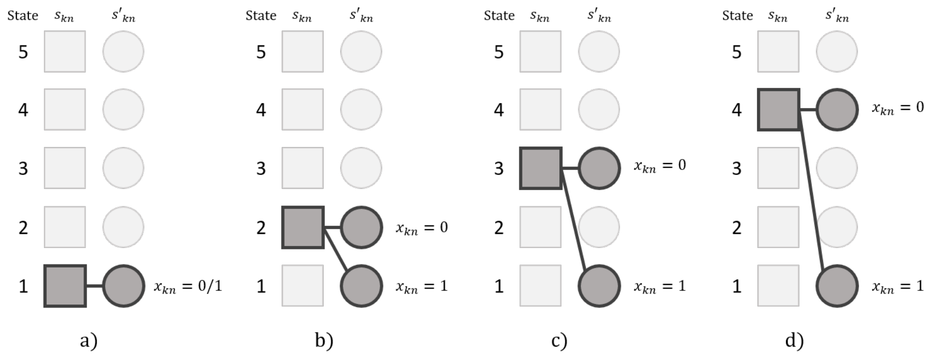


Figure 5.3: The figure shows possible transition between states in a stage due to maintenance, for a component represented by five discrete states. Note that maintenance is only possible following an inspection ($y_n = 1$).

Following maintenance, a component may deteriorate. P_{ijk} indicates the probability of component $k \in |\mathcal{K}|$ deteriorating from state $i \in |\mathcal{S}|$ to state $j \in |\mathcal{S}|$. This is the transition between the pre-deterioration state ($s_n^{'}$) and the pre-maintenance state of the next stage ($s_{(n+1)}$). If the component does not deteriorate, it remains in the same state, given by the probability P_{iik}. If a component fails it first goes to the maximum state, $|\mathcal{S}|$, but as it is repaired immediately, it ends up in state 1. We describe this deterioration in Equation 5.2.

$$s_{k(n+1)} = \begin{cases} s_{kn}^{'} & \text{with probability } P_{iik} \mid s_{kn}^{'} = i \quad i \in S \setminus |S| \\ j & \text{with probability } P_{ijk} \mid s_{kn}^{'} = i, \; i < j, \quad i, \; j \in S \setminus |S| \\ 1 & \text{with probability } P_{i|S|k} \mid s_{kn}^{'} = i \quad i \in S \setminus |S| \end{cases}, \; k \in \mathcal{K}, \; n \in \mathcal{N} \tag{5.2}$$

In Figure 5.4, we illustrate the possible transitions from the pre-deterioration state in stage n to the pre-maintenance state in stage $n+1$ due to deterioration and failure. In Figure 5.4 a) we see that the state can change to any other state. It is worth to note that when going to the failed state, state 5 in the illustration, the component is then repaired and ends up in state 1. Failing in state 1, which means going to state 1 in the next stage, must be considered differently than not deteriorating and remaining in state 1, as failing also incurs the cost of failure. The probabilities for each transition are indicated to the right of the decision nodes in $n+1$.

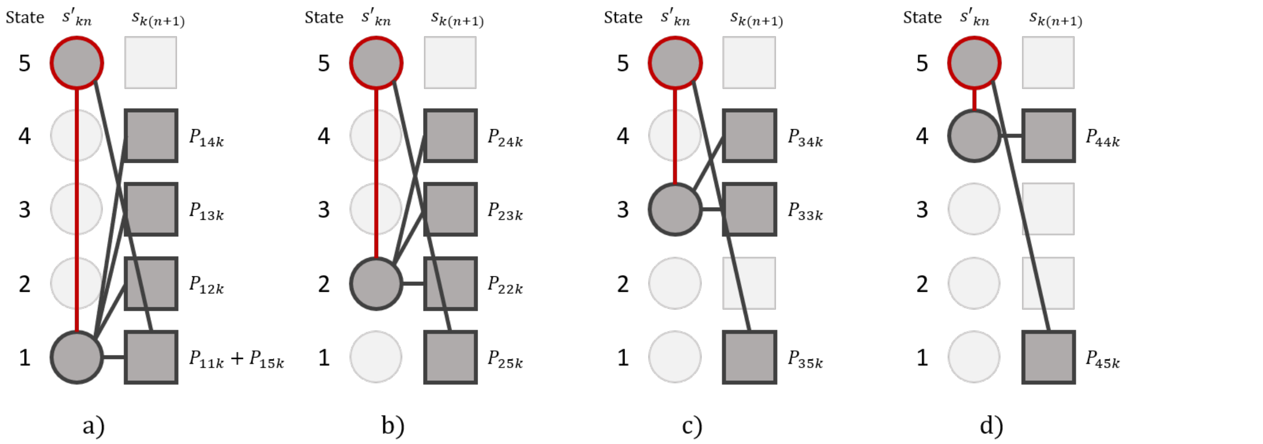


Figure 5.4: The figure shows the possible transition between states in a stage due to deterioration for a component represented by five discrete states.

We see that the probability of ending up in state 1 in Figure 5.4 a) is the combination of P_{11k} and P_{15k}, while the probability of ending up in state 1 in b) is only given by P_{25k}, in c) by P_{35k} and in d) by P_{45k}. We also see that the number of possible transitions is fewer when the pre-deterioration state is 2 compared to 1, 3 compared to 2 and so on. This is because the state cannot improve unless the component is maintained or repaired following a failure.

All the transition probabilities, P_{ijk}, are summarized in a deterioration matrix for each component, D_k. The first row represents the transition probabilities from the first state to the rest and so forth. It can be described as follows:

$$D_k = \begin{bmatrix} P_{11k} & P_{12k} & \dots & P_{1(|S|-1)k} & P_{1|S|k} \\ 0 & P_{22k} & \dots & P_{2(|S|-1)k} & P_{2|S|k} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & P_{(|S|-1)(|S|-1)k} & P_{(|S|-1)|S|k} \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \tag{5.3}$$

Recursion function

We want to find the inspection interval, z, and the maintenance policy, x, that minimize the expected cost over the planning period. As we use backwards recursion, the function's value in the first stage equals the total expected costs for the entire planning horizon. Our objective function thus becomes:

$$\min_{z} f_1(s_1, z) \tag{5.4}$$

where:

$$f_n(s_n,\ z) = \min_{x_n}\{$$

$$\sum_{k=1}^{|\mathcal{K}|} x_{kn} \cdot C_k^{MV} + (1 - \prod_{k=1}^{|\mathcal{K}|}(1 - x_{kn})) \cdot C^{MF} + y_n \cdot C^I$$

$$+ \sum_{k=1}^{|\mathcal{K}|} \sum_{i=1}^{|\mathcal{S}|-1} Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn}) \cdot (P_{i|S|k}) \cdot (C^F + C_k^{MV} + C^{MF} + f_{n+1}(s_{k(n+1)} = 1,\ z)) \tag{5.5}$$

$$+ \sum_{k=1}^{|\mathcal{K}|} \sum_{i=1}^{|\mathcal{S}|-1} \sum_{j=i}^{|\mathcal{S}|-1} Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn}) \cdot P_{ijk} \cdot f_{n+1}(s_{k(n+1)} = j,\ z)\}$$

$$n \in \mathcal{N}$$

The recursion function 5.5 can be broken down as follows:

$$\sum_{k=1}^{|\mathcal{K}|} x_{kn} \cdot C_k^{MV} + (1 - \prod_{k=1}^{|\mathcal{K}|}(1 - x_{kn})) \cdot C^{MF} + y_n \cdot C^I \tag{5.6}$$

Equation 5.6 first sums the variable cost of maintenance for all maintained components. Then it adds the fixed maintenance cost, incurred once if at least one of the components are maintained. Finally, it adds the inspection cost if an inspection takes place.

$$\sum_{k=1}^{|\mathcal{K}|} \sum_{i=1}^{|\mathcal{S}|-1} \sum_{j=i}^{|\mathcal{S}|-1} Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn}) \cdot P_{ijk} \cdot f_{n+1}(s_{k(n+1)} = j,\ z) \tag{5.7}$$

Equation 5.7 is the expected present and future costs given failure. For all components and all possible states, we multiply the probability of the component being in that state, given the maintenance decision, $Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn})$, with the probability of failing in that state, $P_{i|S|k}$. This joint probability is first multiplied with the cost from failure, including the forced repair cost and the CENS. Then, the equation multiplies the probability with the expected future costs of the state being 1 in the next stage, as this is the consequence of forced repair. We know the value of $Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn})$

with certainty, being either 0 or 1, based on the values of s_{kn} and x_{kn}, and the transition function given by Equation 5.1. While it does not represent any probabilistic event, we use this notation for mathematical convenience.

$$\sum_{k}^{K}\sum_{i}^{S\backslash|S|}\sum_{j}^{S\backslash|S|} Pr(s'_{kn} = i \mid s_{kn},\ x_{kn}) \cdot P_{ijk} \cdot f_{n+1}(s_{k(n+1)} = j,\ z) \tag{5.8}$$

Equation 5.8 is the expected future costs given the maintenance decision and deterioration outcomes. Similar to Equation 5.7, we look at the probability of being in a given state, and the probability of going to a certain state from the given state. The probability is used to correctly weigh the expected future costs of possible states in the next stage.

Boundary conditions:

$$s_{k1} = B_k, \quad k \in \mathcal{K} \tag{5.9}$$

$$z \leq z^{max} \tag{5.10}$$

$$f_n(s_{kn} = i) \ = \ C^{end}_{ik}, \quad n = |\mathcal{N}| + 1,\ i \in \mathcal{S},\ k \in \mathcal{K} \tag{5.11}$$

$$(y_n \cdot n - 1)\ mod\ z = 0, \quad n \in \{1,\ ...,\ |N|\} \tag{5.12}$$

$$y_n + z \geq 2, \quad n \in \{1,\ ...,\ |N|\} \tag{5.13}$$

$$x_{kn} \leq y_n, \quad k \in \mathcal{K}\ ,\ n \in \mathcal{N} \tag{5.14}$$

The first constraint, 5.9, sets the initial state of the system, given as a parameter from the user. 5.10 forces the decision variable z to be at most z^{max}. Boundary condition 5.11 sets the end of horizon cost for the recursion function. 5.12 ensures that the inspection variable, y_n, becomes 1 every z^{th} period, and zero in the rest for all z except $z = 1$, starting with an inspection in the first stage. This is based on the assumption that we know the initial state of the system, indicating an inspection in the first stage. 5.13 ensures inspection in every stage if $z = 1$. Finally, 5.14 makes sure that maintenance only takes place following an inspection.

5.4 Illustrative example of the model

In this section, we study two simple cases to illustrate the mechanics of the PSDP. First, we present the data used in both cases in section 5.4.1. Then, in Section 5.4.2, we look at a single-unit system, a system with only one component, with relatively fewer states and stages than realistic instances. We look at the resulting policies from the model and calculate some of the expected costs by hand, before exploring the transition between stages in Section 5.4.3. Then we expand the problem somewhat in Section 5.4.4 by introducing a second component to highlight the interdependencies between components.

5.4.1 Example case parameters

Our first example is a system with only one component, that can be in three different discrete states. We look at the system over a planning horizon of 5 years, each year represented by one stage. Our second example is similar to the first, but introduce a second component with different properties and solves for a system of two components.

Case	$\lvert\mathcal{K}\rvert$	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{N}\rvert$	z^{max}	s_1	C^I	C_k^{MV}	C^{MF}	C^F
Single-unit	1	3	5	2	[1]	5	[6]	4	30
Multi-unit	2	3	5	2	[1, 1]	5	[6, 5]	4	30

Table 5.1: Illustrative case parameters

Table 5.1 summarize the parameters that are used in both example cases. Table 5.2 gives the components' deterioration matrices. We use D_1 for the single-unit example, and both for the multi-unit example.

$$D_1 = \begin{bmatrix} 0.5 & 0.4 & 0.1 \\ 0.0 & 0.6 & 0.4 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \qquad D_2 = \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.0 & 0.8 & 0.2 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

Table 5.2: Deterioration probabilities for component 1 and component 2 in D_1 and D_2, respectively

5.4.2 Single-unit example

When solving the single-unit example, we calculate the expected cost of every possible state-decision combination in each stage. In this thesis, we use the term *stage table* when referring to the tables that hold the calculated values for all state-decisions. We use these values for each stage to find the optimal choice for any state of the system. The collection of these optimal choices is the optimal policy, i.e. the model's solution to the problem. Figure 5.5 shows the resulting stage tables along with optimal values for each stage from the PSDP, where the optimal inspection interval from the solution is $z = 2$.

$n = 1$

$x_1 \setminus s_1$	1	2
0	59.0	
1	69.0	
$x_1^*(s_1)$	0	
$f(s_1)$	59.0	

$n = 2$

$x_2 \setminus s_2$	1	2
0	44.4	58.4
1		
$x_2^*(s_2)$	0	0
$f(s_2)$	44.4	58.4

$n = 3$

$x_3 \setminus s_3$	1	2
0	36.4	51.2
1	46.4	46.4
$x_3^*(s_3)$	0	1
$f(s_3)$	36.4	46.4

$n = 4$

$x_4 \setminus s_4$	1	2
0	21.8	35.8
1		
$x_4^*(s_4)$	0	0
$f(s_4)$	21.8	35.8

$n = 5$

$x_5 \setminus s_5$	1	2
0	13.8	28.2
1	23.8	23.8
$x_5^*(s_5)$	0	1
$f(s_5)$	13.8	23.8

Endtable	1	2
0		
1		
End of horizon cost	0	12

Figure 5.5: Stage tables returned by the PSDP for all stages in the planning horizon

From Section 5.3 we know that the PSDP uses backwards recursion in its calculations, i.e. it starts calculating all the values for the final stage table and then uses those values to calculate the values for the stage table in $n = 4$ and so on. When calculating the stage table in the final stage ($n = 5$), our implementation use what we define as an *endtable*, holding the end of horizon costs.

We start by calculating the values in the stage table where $n = 5$, using the recursion function given by Equation 5.5, in the previous section. Considering the upper-left cell, we have $s_5 = 1$ and $x_5 = 0$. We apply these values, the parameters defined in the problem and the end of horizon cost to Equation 5.6*:

$$\begin{aligned}&\sum_{k}^{K} x_{kn}\cdot C_k^{MV} + (1-\prod_{k}^{K}(1-x_{kn}))\cdot C^{MF} + y_n\cdot C^I\\&= 0\cdot 6 + (1-(1-0))\cdot 4 + 1\cdot 5\\&= 0+0+5\\&= 5\end{aligned} \tag{5.6*}$$

As there is an inspection in stage 5, but no maintenance takes place, we see that the total costs from maintenance and inspection must be 5, which we find by using our numbers in 5.6. Then calculating the expected cost of failure we use 5.7:

$$\begin{aligned}&\sum_{k}^{K}\sum_{i}^{S\setminus|S|} Pr(s'_{kn}=i \mid s_{kn},\ x_{kn})\cdot(P_{i|S|k})\cdot((C^F + C_k^{MV} + C^{MF}) + f_{n+1}(s_{k(n+1)}=1,\ z))\\&= 1\cdot 0.1\cdot(30+6+4+0) + 0\cdot 0.4\cdot(30+6+4+0)\\&= 4+0\\&= 4\end{aligned} \tag{5.7*}$$

For this example, we only sum over one component in 5.7. The probability of $s'_5 = 1$ is 1 given $s_5 = 1$ and $x_5 = 0$, and the probability of $s'_5 = 2$ is consequently 0. Looking at D_1 we see that the probability of deteriorating to the failure state from state 1, P_{131}, is 0.1 (the upper-right value in D_1). This probability is multiplied by the cost of failure and the expected future cost of being in state 1. The cost of failure is the combination of CENS, component maintenance cost and fixed maintenance cost, summing to 40. The expected future cost is in this case the end of horizon cost of state 1, being 0. Finally, we calculate the expected future costs for all the events where the component does not fail:

$$\begin{aligned}&\sum_{k}^{K}\sum_{i}^{S\setminus|S|}\sum_{j}^{S\setminus|S|} Pr(s'_{kn}=i \mid s_{kn},\ x_{kn})\cdot P_{ijk}\cdot f_{n+1}(s_{k(n+1)}=j,\ z)\\&= (1\cdot 0.5\cdot 0 + 1\cdot 0.4\cdot 12) + (0\cdot 0\cdot 0 + 0\cdot 0.6\cdot 12)\\&= (0+4.8)+(0+0)\\&= 4.8\end{aligned} \tag{5.8*}$$

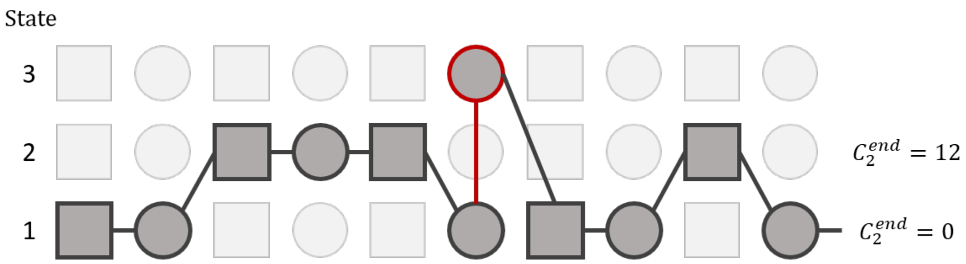
Summing all the parts of the recursion function we end up with:

$$5 + 4 + 4.8 = 13.8$$

This is the same value that the model returns. We suggest that the reader does a calculation to verify the results.

5.4.3 Applying the results from the single-unit example

Figure 5.6 illustrates applying the policy given by the PSDP on a possible scenario considering the single-unit system in the example.



1	1	2	2	3	3	4	4	5	5	Stage (n)
1	-	0	-	**1**	-	0	-	**1**	-	Inspection (y_n)
0	-	-	-	**1**	-	-	-	**1**	-	Maintenance (x_{kn})
-	**1**	-	0	-	**2**	-	**1**	-	0	Deterioration (Δs)
-	0	-	0	-	**1**	-	0	-	0	Failure
5	0	0	0	15	40	0	0	15	0	Incurred cost
59.0	-	58.4	-	46.4	-	21.8	-	23.8	-	Recursion function $f_n(s_{kn})$

Figure 5.6: A possible scenario for the single-unit system over the planning horizon.

Starting in the first stage (i.e. from the left), we do not maintain the component in the first stage. This is in accordance with the optimal choice given by the first stage table in figure 5.5, where $x_1^*(1) = 0$. We see that the component deteriorates between stage 1 and stage 2. As there is no inspection in the second stage, no choice for maintenance can be made, and the state remains 2. Between stage 2 and 3 there is no deterioration. In stage 3, the system is inspected, and according to the policy from the PSDP the component is maintained. Despite our efforts to avoid failure by maintaining the component, we are unlucky, and the component fails in stage 3. As the component is repaired, the state is 1 in stage 4. Again, there is no inspection, thus no available maintenance decision. Then the component deteriorates between stage 4 and 5, and in stage 5, the component is maintained. As the component does not deteriorate in stage 5 (the last node), the incurred end of horizon cost is 0.

We recognise the first recursion function value, 59.0, from the first stage table in figure 5.5. The recursion function value represents the expected total costs from the current and future states. Thus this first recursion function value represents the total expected costs over the planning horizon. Similarly to the first value, we recognise the other recursion function values from the other stage tables, depending on the state in that stage, e.g. in the second table, we have the second value of 58.4 associated with being in state 2, rather than the first value of 44.4 associated with state 1. The recursion function values are typically decreasing, as the stages with expected future costs become fewer with time. Thus, the expected future costs decrease with time. However, we see that the recursion function value of stage 5 is slightly higher than that of stage 4. This is because the state in stage 5 is worse than the state in stage 4, having higher expected costs.

Briefly looking at the costs we see that inspection without maintenance incurs a cost of 5, inspection and maintenance cost 15, and a failure costs 40. The total costs from this scenario sums to 75. This is higher than the expected cost of 59, but then again, we did get unlucky when getting a failure directly after maintenance.

5.4.4 Multi-unit example

We now briefly look at an extension to the previous example, adding one more component to the system. Let the component from the previous example have index 1, and the new component have index 2. The example uses the same parameters as previously introduced, presented again in Table 5.3 with the deterioration probabilities in Table 5.4.

Case	$\lvert\mathcal{K}\rvert$	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{N}\rvert$	z^{max}	s_1	C^I	C_k^{MV}	C^{MF}	C^F
Multi-unit	2	3	5	2	[1, 1]	5	[6, 5]	4	30

Table 5.3: Illustrative case parameters for the multi-unit example

$$D_1 = \begin{bmatrix} 0.5 & 0.4 & 0.1 \\ 0.0 & 0.6 & 0.4 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \qquad D_2 = \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.0 & 0.8 & 0.2 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

Table 5.4: Deterioration probabilities for component 1 and component 2 in D_1 and D_2, respectively

Solving the PSDP for a multi-unit problem mostly follows the same procedure as for a single-unit problem. We will not go through the multi-unit example to the same extent, but rather point out a few important distinctions. The stage tables for the multi-unit example are found in Figure 5.7. We first note that each table is significantly larger. This should not be too surprising, as the number of possible system states grows exponentially with the number of components in a problem.

$n = 1$

$x_1 \backslash s_1$	11	12	21	22
00	86.6			
01	95.6			
10	96.6			
11	101.6			
$x_1^*(s_1)$	00			
$f(s_1)$	86.6			

$n = 2$

$x_2 \backslash s_2$	11	12	21	22
00	66.5	74.8	80.3	88.2
01				
10				
11				
$x_2^*(s_2)$	00	00	00	00
$f(s_2)$	66.5	74.8	80.3	88.2

$n = 3$

$x_3 \backslash s_3$	11	12	21	22
00	53.1	61.9	67.9	76.6
01	62.1	62.1	76.9	76.9
10	63.1	71.9	63.1	71.9
11	68.1	68.1	68.1	68.1
$x_3^*(s_3)$	00	00	10	11
$f(s_3)$	53.1	61.9	63.1	68.1

$n = 4$

$x_4 \backslash s_4$	11	12	21	22
00	33.1	41.4	46.9	54.8
01				
10				
11				
$x_4^*(s_4)$	00	00	00	00
$f(s_4)$	33.1	41.4	46.9	54.8

$n = 5$

$x_5 \backslash s_5$	11	12	21	22
00	19.6	29.6	34.1	44.0
01	28.6	28.7	43.1	43.1
10	29.6	39.6	29.7	39.6
11	34.6	34.7	34.7	34.7
$x_5^*(s_5)$	00	01	10	11
$f(s_5)$	19.6	28.7	29.7	34.7

Endtable	11	12	21	22
00				
01				
10				
11				
End of horizon cost	0	10	12	22

Figure 5.7: The figure shows the stage tables returned by the PSDP for all stages in the planning horizon for a two-component case.

We see that the decisions related to the first component are the same as for the single-unit example. If the state is 1 ($s_{1n} = 1$), there should be no maintenance ($x_{1n}^*(s_{1n} = 1) = 0$), and if the state is 2 the component is maintained ($x_{1n}^*(s_{1n} = 2) = 1$) for stages 3 and 5, having inspections. Then, we note that the same is not always the case for the second component. Looking at the third stage table ($n = 3$), we see that if only the second component is in state 2 ($s_3 = 12$), that is $s_{13} = 1$ and $s_{23} = 2$), no maintenance is suggested ($x_3^*(s_3 = 12) = 00$). However, if both components are in state 2, maintenance is suggested for both components ($x_3^*(s_3 = 22) = 11$). This highlights the dependencies between components, i.e. the second component will only be maintained if it can share the fixed maintenance cost with the first component.

If we look at the fifth table ($n = 5$), we see that the model now suggests maintenance for the second component regardless of the first component. This is due to the end of horizon cost making maintenance slightly more preferable than no maintenance, costs being 28.7 and 29.6, respectively.

A final note on the multi-unit example is the calculation of probabilities. The probabilities of changing to different states depend on all components' deterioration matrices, making the calculation of the transition probabilities slightly more complex for larger-sized problems. E.g. the probability of going from $s_n = 11$ to $s_n = 21$ is given by $P_{121} \cdot P_{112} = 0.4 \cdot 0.7 = 0.28$. We encourage the reader to calculate a few of the table-values by themselves to better understand how the model works.

5.5 Sequential SDP model

In the Sequential Stochastic Dynamic Program (SSDP or Sequential SDP), the time between inspections are no longer set with fixed intervals but chosen following an inspection based on the revealed information and chosen maintenance decision. In accordance with maintenance optimization literature presented in Chapter 3, we call this inspection structure "sequential inspection". This structure allows the model to better adapt to the revealed information, but with a larger number of possible decisions at each stage, and consequently an increase in the number of required calculations.

5.5.1 Additional model assumptions

In addition to the assumptions shared by both models, the Sequential SDP model has the following assumption:

- **The decision maker decides the time until the next inspection when carrying out an inspection.** The time until an inspection is not decided at every stage, only those in which an inspection takes place.

5.5.2 Additional notation

The notation of the Periodic SDP still applies, with an exception outlined underneath.

Decision variables:

z_n	$z_n \in \{1, \ ..., \ \lvert z^{max} \rvert\}$. The number of stages until the next inspection. If there is an inspection in the next stage, the value is 1. In the stage where an inspection takes place, z_n is set to a new value and will never become zero.

We introduce z_n as a replacement to z in the PSDP. The optimal choice, z_n^*, is the z_n that minimizes expected future costs, given the maintenance decision x_n^*. This decision can only be taken in stages where an inspection takes place ($y_n = 1 \ \equiv \ z_{n-1} = 1$), otherwise it is forced to be one less than the previous time until inspection-value ($z_n = z_{n-1} - 1$). This is shown in Equation 5.15:

$$z_{(n)} = \begin{cases} \mathrm{z}_n^* & \text{if } y_1 = 1 \\ \mathrm{z}_{n-1} - 1 & \text{otherwise} \end{cases}, \ n \in \mathcal{N} \tag{5.15}$$

5.5.3 Model formulation

Recursion function

The recursion function remains fairly similar, with one important distinction. We now seek to minimize z_n at each stage. The objective becomes:

$$\min f_1(s_1,\ z_0) \tag{5.16}$$

where:

$$\begin{aligned}
f_n(s_n,\ z_{n-1}) = \min_{x_n,\ z_n} \{ \\
\sum_{k=1}^{|\mathcal{K}|} x_{kn} \cdot C_k^{MV} + (1 - \prod_{k=1}^{|\mathcal{K}|}(1 - x_{kn})) \cdot C^{MF} + y_n \cdot C^I \\
+ \sum_{k=1}^{|\mathcal{K}|} \sum_{i=1}^{|\mathcal{S}|-1} Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn}) \cdot (P_{i|S|k}) \cdot (C^F + C_k^{MV} + C^{MF} + f_{n+1}(s_{k(n+1)} = 1,\ z_n)) \\
+ \sum_{k=1}^{|\mathcal{K}|} \sum_{i=1}^{|\mathcal{S}|-1} \sum_{j=i}^{|\mathcal{S}|-1} Pr(s_{kn}^{'} = i \mid s_{kn},\ x_{kn}) \cdot P_{ijk} \cdot f_{n+1}(s_{k(n+1)} = j,\ z_n)\},\ n \in \mathcal{N}
\end{aligned} \tag{5.17}$$

Note that the expected future cost, f_{n+1}, now takes in z_n as an argument, rather than z like the PSDP. Other than that, the recursion function remains the same.

Additional boundary conditions:

$$z_0 = 1 \tag{5.18}$$

$$y_n \cdot (z_{(n-1)} - 1) = 0, \quad n \in \mathcal{N} \tag{5.19}$$

$$y_n + z_{(n-1)} \geq 2, \quad n \in \mathcal{N} \tag{5.20}$$

Boundary conditions 5.18, 5.19 and 5.20 replace conditions 5.12 and 5.13 from the previous section. Boundary condition 5.18 is set so that an inspection takes place in the first stage. 5.19 ensures that the variable indicating inspection, y_n, does not take the value 1 unless it follows a stage where $z_{(n-1)}$ is 1, and condition 5.20 ensures that y_n is 1 following a stage where $z_{(n-1)}$ is 1.

Chapter 6

Case Study

In this chapter, we present the case data that is used to test the mathematical models from Chapter 5. The cases are based on data and insights provided by our case company, Wiseline AS. They are designed with the intention to show practical implications of power grid inspection and maintenance optimization using stochastic dynamic programming. Section 6.1 gives a brief introduction to the case company, while Section 6.2 presents the parameters considered in the case study. In Section 6.3 we define the Base Case, several variations to this Base Case and the cases used for run time analysis. Section 6.4 defines conventional policies, similar to those used by power grid operators today. These are used to compare the policies returned from the mathematical models, studied further in the Computational Study.

6.1 The case company: Wiseline AS

Wiseline AS is a Norwegian company that provides decision support to power grid operators. Using lifetime analysis and system condition data, they simulate and visualize the grid's development over time, and the implications of using different inspection and maintenance strategies. For instance, they have often seen that grid operators can make significant savings by adopting less conservative maintenance strategies, without unacceptable increase in risk of failure. As part of their work, Wiseline assist power grid operators to better capture and structure data about the grid, allowing for more comprehensive data analysis in an industry where conventional methods typically play a central role. The insights are useful on a strategic level, understanding the long-term implications of different inspection and maintenance policies. It is also valuable at a tactical level, e.g. deciding which lines should be prioritized in a given year. Finally, it can be used on an operative level, deciding which maintenance thresholds should be used for different mast components.

While Wiseline can provide new insights for the power grid operators through data analysis, they currently do not find optimal policies. Simulated policies are typically set in workshops with power grid operators, and are based on convention and gut feel. They point to optimization as an interesting topic to explore further to enhance decision support for power grid operators.

6.2 Case parameters

In this section, we briefly discuss the mathematical models' parameters and how they are found. Most of the case data used in this thesis are based on insights and real-life data provided by Wiseline, applied to the problem context. We make several assumptions, and the results should be considered illustrative.

The utility mast and its components

As discussed in Chapter 2, a power grid line consists of several utility masts. Masts in the same line share many properties, such as related costs and implications of failure. Typically, many of the masts are located in similar environments, and a common assumption is to say they experience similar deterioration processes. Thus, deriving the optimal maintenance and inspection policies for one representative utility mast will provide valuable insights when considering the maintenance and inspection strategy for the entire line.

A utility mast has several components. Components serve different functions, such as electricity transmission, structural support and safety, to mention a few. When considering maintenance policies for a mast's components, some will have more straightforward policies than others. For example, a component that is always maintained when found in a given state, regardless of other components states, has a more straightforward policy than one where the maintenance decision is linked to the state of other components in the system.

In our models, we focus on a subset of the mast's components, where all modelled components are critical to the mast's ability to deliver electricity. That is, any component failure leads to mast failure. Furthermore, we will primarily look at components with non-trivial maintenance policies (i.e. policies with several economic dependencies). We use the term *system* when referring to our selection of components that represent the mast.

Figure 6.1 shows the mast modelled in the cases with all component IDs and names. The mast consists of four components that the mast requires to function. These are chosen to give a simplified, representative development of a mast's deterioration along a planning horizon. Unless explicitly stated, these components are all included in the cases presented in this chapter.

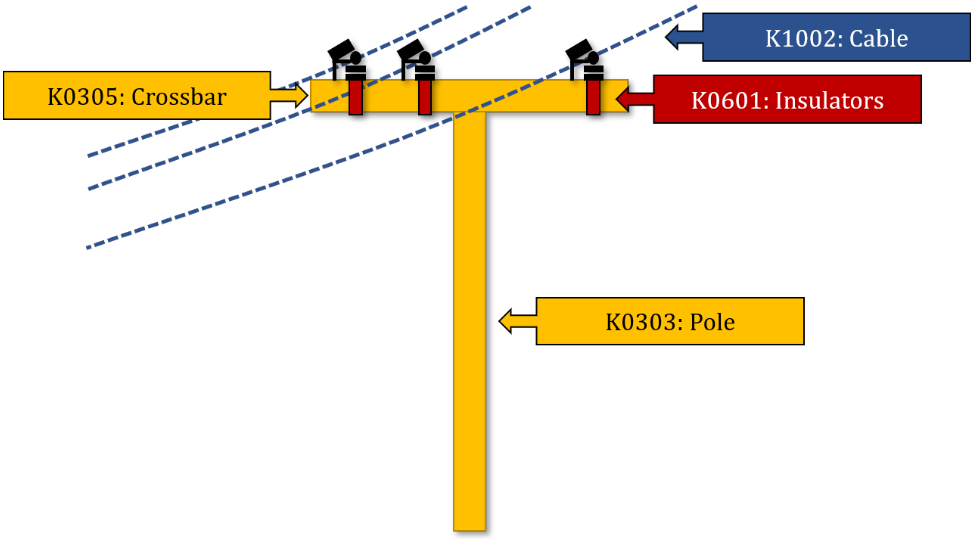


Figure 6.1: The figure shows the utility mast modelled for our case study. All the components on the form are critical and intended to collectively represent a simplified utility mast.

Deterioration

Different components deteriorate differently, with deterioration probabilities represented in a matrix for each component. Such a matrix is based on empirical data, physical deterioration models or lifetime expectancy models. We assume that the user of the models provides the deterioration matrices. A deterioration matrix states the probability of a component transitioning from one state to another, for all possible states.

Component maintenance cost

Each component has an associated maintenance cost. After a component is maintained it is considered "good as new", implying that maintenance of a component means replacing it. The cost of maintenance is comprised of the material costs of the component itself and the cost from a technician's time spent on maintaining the component. It does not include the cost of mobilization and travel to the location, which is instead captured by the fixed maintenance cost.

Fixed maintenance cost

The fixed maintenance cost is incurred whenever at least one component is maintained or repaired after a failure. It represents the cost of preparing, mobilizing and travelling to the location. The cost is primarily driven by the time used by the team of technicians, and their hourly rate assumed to be 1000 NOK per technician. The time used depends on factors such as proximity of the mast to maintenance base and how easy it is to get to the mast's location.

Cost of inspection

Different inspection types have different costs, but for the cases in this thesis, we consider the thorough inspection that returns the actual state of each of the components in the modelled mast. Inspections are done manually, and we use the same hourly rate as with fixed maintenance, 1000 NOK, as the required labour type is similar. The time used on an inspection also depends on the same factors as the fixed maintenance cost, as we assume that all components are inspected when an inspection is carried out. However, we assume that inspection requires less time because of the additional preparation, mobilizing and the number of technicians required by any maintenance.

Cost of failure

The cost of failure is incurred whenever a component fails and causes system failure. The cost is the cost of energy not supplied (CENS) combined with the cost of repairing the failed component. Even though components can fail in the same year, they rarely fail at the same time. Thus each component failure incurs a failure cost. The cost of repair is the fixed maintenance cost and the maintenance cost of the failed component. Wiseline indicates that CENS vary widely based on the negative societal impact caused by failure, and rates range between 447 NOK per hour to 98 117 NOK per hour. The cases we study in this thesis have a CENS based on a rate within this interval, multiplied by an estimated downtime.

End of horizon costs

At the end of the planning horizon, the masts value is most practical purposes dependent on its state at that point. For example, a system where all components are in state 1 will typically be more worth than a system where all components are in state 3. However, other factors can influence the value at the end. For instance, if a line will no longer be used, or is subject to a complete re-installation at the end of the planning horizon, the state does no longer matter. The operator specifies the cost of ending up in a state, the *end of horizon cost*, for all the possible states of each component.

Planning horizon

The models and simulation use a stage structure to model time. For all cases in this thesis, each stage represents one year. A maintenance and inspection plan typically consider several decades. This is natural as the grid operators often inspect thoroughly when required by the regulations, every tenth year. Although the optimal inspection interval may vary from this, it is reasonable to consider between 20-50 years to obtain a lasting maintenance plan.

Initial states

The initial states of the components are required to model and simulate the system. We assume the components to be "good as new", i.e. $s_1 = [1, \ ..., \ 1]$, unless something else is explicitly stated. More often than not, this cannot be assumed, and the mathematical models can still be applied when using other initial states.

Inspection interval

Both mathematical models solve for optimal inspection intervals, albeit in different ways. The intervals are both restricted upwards by z^{max}. This constraint can be viewed as the regulation stating the required inspection frequency of thorough inspections of the power grid. We mostly use $z^{max} = 10$ in this thesis, being the regulation imposed on the Norwegian power grid for thorough inspections.

6.3 Overview of the cases

In this section, we present the cases used to analyze the results from the Periodic and Sequential SDPs. We first present a *Base Case* with parameters based on data provided by Wiseline. Then we introduce several variations to the Base Case, used to see how changes in parameters influence the policies behaviour and performance. Finally, we outline some cases used for run time analysis.

6.3.1 The Base Case

The Base Case considers a utility mast that is part of a 22 kV power grid line. We consider a system of four components on this mast, and all are viewed as critical to the system. The components with associated maintenance costs and end of horizon costs (for states 1 to 4, respectively) are found in Table 6.1. Each component's name and ID correspond to the components specified in Figure 6.1.

Component ID	Component	C_k^{MV}	C_{ik}^{End}
K0303	Pole	23.00	[0, 19.50, 29.00, 29.00]
K0305	Crossarm	3.50	[0, 4.75, 9.50, 9.50]
K0601	Insulators	0.82	[0, 3.41, 6.82, 6.82]
K1002	Cable	5.20	[0, 5.60, 11.20, 11.20]

Table 6.1: The components used in the Base Case and their variable maintenance costs. Values are in kNOK.

In table 6.2 we find the planning horizon, maximum inspection interval, possible component states, initial state and costs used for the Base Case. The intuition of the values is discussed in Section 6.2. The planning horizon, $|\mathcal{N}|$, is 50 years. We use a maximum inspection interval, z^{max} of 10, as is given by regulations. All components can be in 5 possible states, and we consider the system to be "good as new" with all components in state 1, i.e. $B_k = 1$ for all components $k \in \mathcal{K}$ at the start of the planning horizon. Inspection is assumed to take two hours, giving an inspection cost, C^I, of 2 000 NOK. We assume that any maintenance requires 6 hours of preparation, mobilizing and travelling from all involved technicians, resulting in a fixed maintenance cost, C^{MF}, of 6 000 NOK. Furthermore, we assume an expected downtime of 2 hours at an hourly CENS-rate of 7 500 NOK/h, resulting in an expected $CENS$ of 15 000 NOK per failure.

Parameter	$\lvert\mathcal{N}\rvert$	z^{max}	$\lvert\mathcal{S}\rvert$	B_k	C^I	C^{MF}	$CENS$
Value	50 years	10 years	5	[1,1,1,1]	2 kNOK	6 kNOK	15 kNOK

Table 6.2: Parameters and additional costs for the Base Case

The deterioration probabilities for all components are summarized in table 6.3. Each row represents the probability of deteriorating from a state equal to the row index to all other states. That is, the upper left number in each matrix represents the probability for staying in state 1, the second number from the left in the top row represents the probability for deteriorating from state 1 to state 2, and so on. The probabilities are based on a deterioration constant for each component, provided by Wiseline, adapted to fit our model formulation. However, they should be viewed as purely illustrative, and stronger empirical data and lifetime analysis are necessary for precise deterioration probabilities.

$$D_{K0303} = \begin{bmatrix} 0.944 & 0.014 & 0.014 & 0.014 & 0.014 \\ 0.000 & 0.892 & 0.036 & 0.036 & 0.036 \\ 0.000 & 0.000 & 0.784 & 0.108 & 0.108 \\ 0.000 & 0.000 & 0.000 & 0.676 & 0.324 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.000 \end{bmatrix}$$

$$D_{K0305} = \begin{bmatrix} 0.920 & 0.020 & 0.020 & 0.020 & 0.020 \\ 0.000 & 0.841 & 0.053 & 0.053 & 0.053 \\ 0.000 & 0.000 & 0.680 & 0.160 & 0.160 \\ 0.000 & 0.000 & 0.000 & 0.520 & 0.480 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.000 \end{bmatrix}$$

$$D_{K0601} = \begin{bmatrix} 0.920 & 0.020 & 0.020 & 0.020 & 0.020 \\ 0.000 & 0.919 & 0.027 & 0.027 & 0.027 \\ 0.000 & 0.000 & 0.920 & 0.040 & 0.040 \\ 0.000 & 0.000 & 0.000 & 0.920 & 0.080 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.000 \end{bmatrix}$$

$$D_{K1002} = \begin{bmatrix} 0.964 & 0.009 & 0.009 & 0.009 & 0.009 \\ 0.000 & 0.964 & 0.012 & 0.012 & 0.012 \\ 0.000 & 0.000 & 0.964 & 0.018 & 0.018 \\ 0.000 & 0.000 & 0.000 & 0.964 & 0.036 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.000 \end{bmatrix}$$

Table 6.3: Deterioration probabilities for each component used in the Base Case

6.3.2 Case variations

In addition to the Base Case, we analyze the implications of varying a number of the case parameters. We consider a system of three components rather than four for all case variations and use a planning horizon of 25 years. This reveals many of the same tendencies, but at a lower run time. The case variations' remaining parameters are the same as for the Base Case, except for those parameters we vary over for every variation. We use components K0303, K0305 and K0601 from the base case for these variations. We will vary over the three parameters C^I, C^{MF} and $CENS$.

Variation in	Start	End	Increment	No. of cases	CaseID
C^I	0	6	0.5	13	VCI_01 - VCI_13
C^{MF}	0	24	2.0	13	VCM_01 - VCM_13
$CENS$	0	120	10.0	13	VCE_01 - VCE_13

Table 6.4: The table shows case variations of the Base Case where one parameter is varied at a time. All numerical values are given in kNOK, except for the number of cases.

The case variations are summarized by Table 6.4. Each row of all the tables states which parameter that is different from the Base Case.

The varied parameters have a "start"-value, an "end"-value, an "increment" and a "caseID". The "start"-value indicates the value of the varied parameter in the first case variation, the "increment" indicates the how much the "start"-value is increased to make a new case variation, and the "end"-value indicates the value of the last case variation, where the increment stops. Each case variation is assigned a number in their "CaseID" in increasing order from the "start"-value to the "end"-value of the varying parameter. The letter combination distinguishes case variations of one set of parameters from another.

In the first row of Table 6.4, we see that the case variations' "CaseID" begins with "VCI", indicating variation in the cost of inspection. The cases have the same parameters as the Base Case, but C^I is varied between 0 and 6. As the increment is 0.5, we know that the case "VCI_04" will have an inspection cost of 1.5.

We will now give a brief summary of the case variation choices for each parameter and the motivation behind these variations.

Varying cost of inspection

The cost of an inspection will vary between different grid lines. Furthermore, new methods of inspections are constantly being developed. Hence the cost of inspection can change over time. Therefore, it is highly relevant to explore how varying inspection cost influences the models' suggestions and performances.

When varying over inspection cost, we start with C^I at 0 NOK going to 6 000 NOK, with 500 NOK increments. Setting the cost to 0 allows us to study behaviour when inspection cost is not a factor.

Varying fixed maintenance cost

Fixed maintenance cost is varied in a similar fashion as the cost of inspection. However, we look at a larger interval, as the costs of transporting necessary equipment to remote locations can be more costly than only transporting personnel. Starting at $C^{MF} = 0$ NOK we increment by 2 000 NOK, ending at $C^{MF} = 24\,000$ NOK.

Varying CENS

The penalty cost of failure, CENS, is highly influential on the economic implications of failure. Hence, it is interesting to study how it changes it affects the models' suggested policies. When looking at variation only in CENS, the interval starts at 0 NOK and ends at 120 000 NOK, with 10 000 NOK increments. Setting CENS to 0 NOK implies a very low failure cost that may result in a "run to failure"-policy. However, failure still results in a forced repair, incurring both component replacement cost and fixed maintenance cost. In some situations, this might make it more advantageous to maintain when several components are in a highly deteriorated state, compared to not doing anything. When increasing CENS, we should expect the mathematical models to suggest more conservative policies to mitigate the total expected costs from failure.

6.3.3 Run Time Cases

In our Computational Study, we analyze the run time development of the two mathematical models. We vary parameters that affect the problem size and consequently, the run time. Cost parameters and deterioration probabilities do not affect either the problem size or the number of required calculations for solving the model. Therefore, these costs and probabilities will not be specified for the Run Time Cases. We will now present two different Run Time cases and their variations.

The General Run Time Case

To limit the computations required to analyze the run time development, the *General Run Time Case* (G-RTC) is smaller than the Base Case, considering only 2 components. This allows certain problem dimensions to be varied to a greater extent when analyzing. Relevant parameters for the G-RTC is summarized in Table 6.5. With 2 components, we significantly reduce the number of needed computations, while still having a multi-unit system. Each component's possible states are the same as the Base Case, while the planning horizon is increased to 30. The planning horizon of the G-RTC is longer than for the case variations because we wish to increase the maximum allowed inspection interval to more than 25 when analyzing its impact on the run time. The maximum allowed inspection interval is also kept the same as in the Base Case.

Parameters	$\|\mathcal{K}\|$	$\|\mathcal{S}\|$	$\|\mathcal{N}\|$	z^{max}
Value	2	5	30	10

Table 6.5: Parameters for the General Run Time Case

Variations of the General Run Time Case

The run time of the G-RTC is analyzed for changes of the planning horizon, the possible states for each component and the maximum allowed inspection interval.

Variation in	Start	End	Increment	No. of cases	CaseID
$\|\mathcal{S}\|$	1	10	1	10	GRTS_01 - GRTS_10
$\|\mathcal{N}\|$	10	100	5	19	GRTN_01 - GRTN_19
z^{max}	1	30	1	30	GRTZ_01 - GRTZ_30

Table 6.6: The table shows case variations of the General Run Time Case where one parameter is varied at a time.

Table 6.6 shows the variations of the G-RTC used for run time analysis. Using the same structure as we present the case variations of the Base Case, we see from the table that each G-RTC-variation will only differ from the G-RTC in at most one dimension. Each case variation has a "CaseID", where the numerical values are set in increasing order based on the varying parameter's value. The letter combination distinguishes cases based on different parameters subject to variation.

Parameters	$\|\mathcal{S}\|$	$\|\mathcal{N}\|$	z^{max}
Value	2	4	2

Table 6.7: Parameters for the Special Run Time Case

The Special Run Time Case

As the number of components in the system greatly affects the mathematical models' required calculations, we create a *Special Run Time Case* (S-RTC) varying over the problem's number of components. The relevant parameters for the S-RTC is summarized in Table 6.7. The states space is reduced to 2 possible states per component and the planning horizon to 4 periods. The maximum allowed inspection interval is reduced to 2. **Variations to the Special Run Time Case**

Table 6.8 summarizes all the variations of the S-RTC used in the run time analysis. We see a total of 7 variations of the S-RTC, considering a system comprised of 1 to 7 components.

Variation in	Start	End	Increment	No. of cases	CaseID
$\|\mathcal{S}\|$	1	7	1	7	SRTC_1 - SRTC_7

Table 6.8: The table shows case variations of the Special Run Time Case where the number of components is varied.

6.4 Bench-mark policies

Along with the policies returned from the periodic stochastic dynamic program and the sequential stochastic dynamic program, we consider four different condition-based maintenance-policies (CBM-policies), and the "run to failure"-policy (RTF-policy). CBMs are a maintenance policy type that we introduce in Chapter 2. When inspection reveals a mast component's state (also referred to as "condition"), it is replaced if that state exceeds a certain threshold. The suggested policies vary in periodic inspection interval, and consequently, the state-threshold for the different components is set to reflect an appropriate risk level for the decision-maker.

We define a benchmark policy with short inspection intervals and high maintenance thresholds, assuming that frequent inspections compensate for a more risky maintenance policy. This policy is denoted as *CBM-short-high* (CBM-S-H). We also set a *CBM-extended-low* (CBM-E-L)-policy based on the opposite approach: Extended (long) inspection intervals, but low maintenance thresholds as the system is rarely inspected. Furthermore, we consider a policy trying to balance inspections and maintenance thresholds, called *CBM-balanced* (CBM-B). We also use the RTF-policy, mentioned in Chapter 2.

Policy	Interval	K0303	K0305	K0601	K1002
RTF	10	-	-	-	-
CBM-S-H	2	4	4	4	4
CBM-B	5	3	3	3	3
CBM-E-L	8	2	2	2	2

Table 6.9: The table shows the inspection interval and state-thresholds of the conventional policies for the components used in the Base Case.

The policies serve as a benchmark to our models, as they can reasonably be applied to power grid lines, and reflect how a power grid operators can design their policies. We derive the fourth CBM-policy from the two dynamic programming models and present it in Chapter 7.

The conventional policies are summarized in Table 6.9. Each column of components denotes the least deteriorated state where the corresponding policy will impose maintenance. For the RTF-policy, state "-" means that no maintenance will happen unless a failure occurs. The interval-column denotes the years between each inspection for the different policies.

Chapter 7

Computational Study

In this chapter, we present computational results from using the Periodic and Sequential stochastic dynamic programs from Chapter 5 to solve the cases presented in Chapter 6. We refer to these programs as the "Periodic SDP" or "PSDP" and "Sequential SDP" or "SSDP", respectively. When collectively referring to the stochastic dynamic programs, we use the term "SDPs" or "models". We simulate the system's behaviour over the planning horizon to analyze the models' and benchmark policies' results. For all simulations in this chapter, we use the Python module "random", with a seed of 10^7.

Section 7.1 presents a run time analysis of both SDPs. In Section 7.2, we discuss the maintenance and inspection policies suggested from both SDPs when used on the Base Case. Section 7.3 compares the policies suggested by our models with conventional policies by simulation, highlighting their relative performance. Section 7.4 looks at how varying the case input parameters influence our models' suggested policies. In Section 7.5, we present a heuristic that divides a problem into several smaller subproblems, solves them by using the SSDP and combine the solutions to one for the larger problem. Finally, Section 7.6 reflects on how the results can be applied for real-life purposes, discussing both possibilities and limitations.

7.1 Run Time Analysis

As the SSDP and the PSDP proposed for the Utility Mast Inspection and Maintenance Problem (UMIMP) are based on the same approach, and only differ in the inspection decision space, they have similar run time developments. In this section, we present a run time development for the two models. First, we specify the models' implementation and the computer system used to solve them. Then, we explain how memory is exploited to reduce the run time. Furthermore, we investigate how the run time develops for the two models when the problem cases increase in size along several dimensions.

For the Base Case, the PSDP takes approximately three hours to solve, while the SSDP require approximately nine and a half hours. The remainder of this section uses the Run Time Cases presented in Chapter 6, to provide better analysis.

7.1.1 Implementation

In this thesis, we implement both suggested models, the plots and simulations using the Python programming language, version 3.7, with object-oriented code. The performance is analyzed on a computer with a Microsoft Windows 10 Education 64-bit operating system. The system's specifications are Intel Core i7-8700 CPU with 3.20 GHz and installed memory (RAM) of 32 GB.

7.1.2 Exploiting memory to reduce run time

The SDPs calculate the expected costs for all possible decisions and all possible states, for each stage in the planning horizon. The complete calculation for our problem is given by the recursion functions for both models in Chapter 5. The relation is more generally described for every state in section 3.5 by Equation 3.4. It is given again here for simplicity:

$$f_n(s_n) = \min_{x_n \in \mathcal{X}_n} \{c(s_n, x_n) + f_{n+1}(s_{n+1})\}$$

From the equation, we see that the optimal expected cost value is obtained by calculating the direct expected cost of each possible situation, for all subsequent states. However, when $f_n + 1(s_n + 1)$ is calculated for a given n, calculating $f_n(s_n)$ requires the exact same calculations in addition to $c(s_n, x_n)$. If the dynamic program stores the values used to calculate $f_n + 1(s_n + 1)$, they can be looked up instead of recalculated for $f_n(s_n)$. This strategy of storing already calculated values for future use to reduce the number of computations when evaluating recursively defined functions is known as tabulation (Bird, 1980).

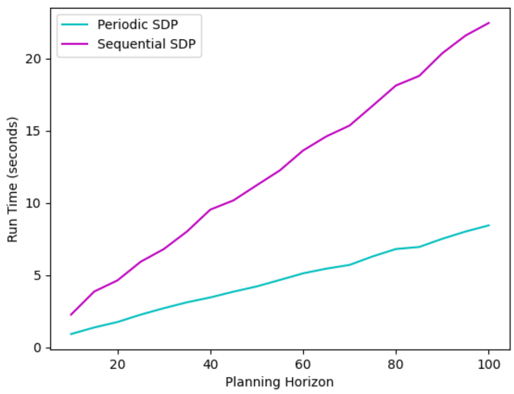
It is important to note that this advantage does not solve the curse of dimensionality, as obtaining the expected cost for each possible state in every stage requires a large number of calculations. Nevertheless, the tabulation strategy does ensure better development in run time when the planning horizon increases.

7.1.3 Run time across different dimensions

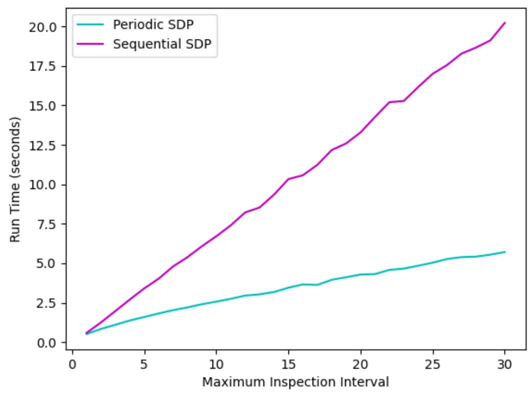
We use variations of the *General Run Time Case* (G-RTC) for analyzing the run time development. The variations depend on which dimension is varied, while all other parameters are kept constant. The only exception is when we look at the run time development for systems with an increasing amount of component. For this, we use the *Special Run Time Case* (S-RTC). The *run time* is defined as the time it takes for both models to obtain all optimal policies for all states in all stages.

Varying planning horizon and maximum inspection intervals

Figure 7.1 shows the run time for both models for variations of the G-RTC, where the planning horizon increases with 5-year increments. As the graph illustrates, the run time increases linearly for both models. This linear development is achieved by the use of tabulation. As states and the number of components are kept constant, each stage requires the same number of calculations. As these are saved during a run, increasing the planning horizon only requires additional calculations for the additional stages.



(a) Varying planning horizon



(b) Varying maximum inspection interval

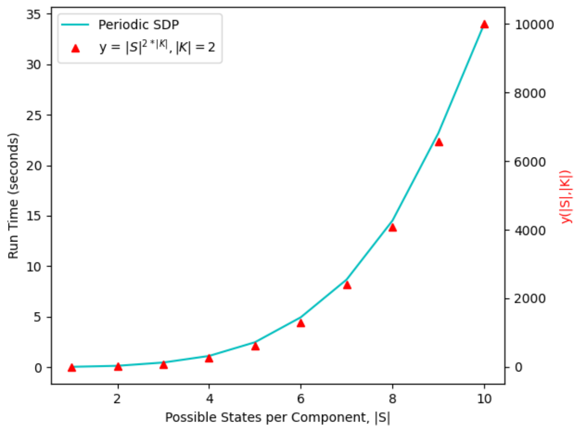
Figure 7.1: The figure shows the run time development with increasing planning horizon or the maximum allowed inspection interval. The cases used in (a) are GRTN_01 - GRTN_19 and the cases used in (b) are GRTZ_01 - GRTZ_30.

The run time development for increasing the maximum allowed inspection interval is also linear for both models, as Figure 7.1b suggests. For the PSDP, this is a natural consequence of calculating optimal policies for each possible interval to find the interval with the lowest expected cost. Increasing the maximum allowed interval means that another policy must be evaluated, given the new possible optimal interval.

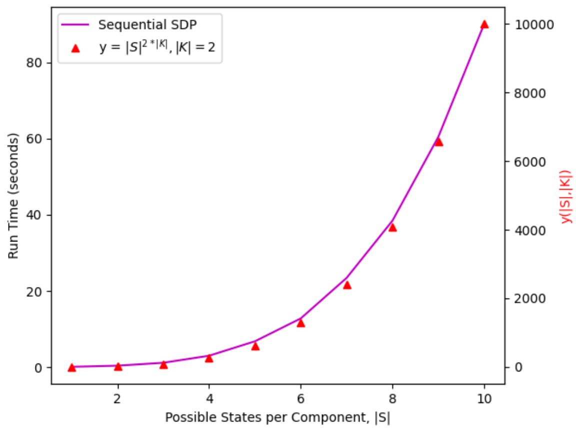
As for the SSDP, the situation with increasing intervals is slightly more complicated. Once again, the tabulation strategy ensures a linearly increasing run time. Because the decision maker may choose between all possible intervals after every inspection, the additional possible decisions must be calculated for every stage when increasing the interval. However, the expected costs of subsequent decisions are still stored and exploited in all preceding stages, as is the case for increased planning horizons.

Varying states per component

Increasing the number of states that each component can be in has two important implications for our models' run time. First, as each of the $|\mathcal{K}|$ components has $|\mathcal{S}|$ individual states, the system can have $|\mathcal{S}|^{|\mathcal{K}|}$ different states, although a failure state immediately forces repair and a new state. Furthermore, we model the possibility of state transitions to all subsequent ("worse" states). This modelling choice means that a component currently in state 1 that can be in $|\mathcal{S}| = 5$ possible states may stay in the same state or transition to all subsequent states in the next stage. That is, a total of 5 transitions are possible. With the same system as described above, the run time complexity for each state transition is bounded by $\mathcal{O}(|\mathcal{S}|^{|\mathcal{K}|})$. As the expected cost in the next stage is calculated for each state by calculating the expected cost of all possible state transitions for all possible states, the number of calculations is bounded by $\mathcal{O}(|\mathcal{S}|^{2|\mathcal{K}|})$.



(a) The Periodic SDP



(b) The sequential SDP

Figure 7.2: The figure shows the run time development for both models on the cases GRTS_01 - GRTS_10 with increasing number of states (|S|) and a polynomial function of the states per component. Note that the number of components (|K|) is kept constant at 2.

The derived expression is plotted with the run time development of the PSDP in Figure 7.2a and the SSDP in Figure 7.2b. We see that the models are of polynomial time with respect to the number of states, even when all state transitions are valid. Furthermore, we use a state space of five discrete states to describe utility mast components. The run time development implies that it may be useful to combine several components and represent them collectively as a sub-system with possibly more than five states to exploit how complexity increases in the stochastic dynamic programs.

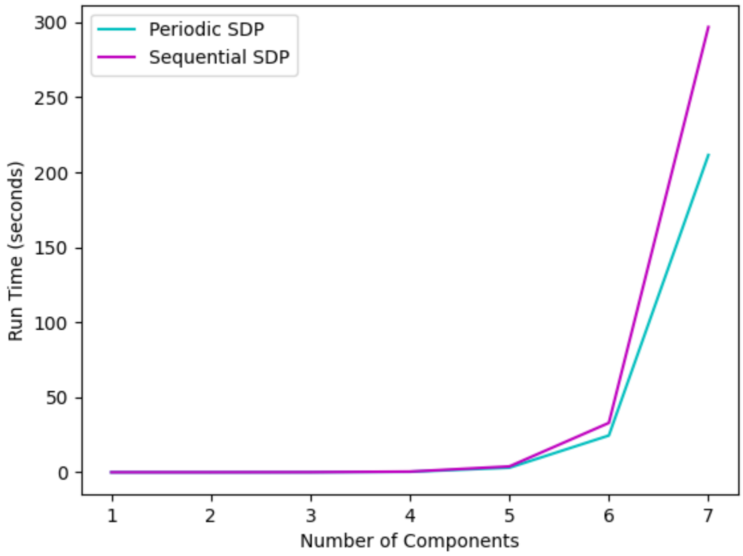


Figure 7.3: The figure shows the exponential run time development with increasing number of components for both models. The used cases are SRTC_1 - SRTC_7.

Varying the number of components

The derived run time complexity of $\mathcal{O}(|\mathcal{S}|^{|\mathcal{K}|})$ implies that the number of components is largely what limits the problem sizes that the models are appropriate for, and thus the models' possible applications. The run time increases exponentially with the number of components, which we can see in Figure 7.3. As suggested when discussing varying states, if one can combine solutions considering sub-systems with fewer components, the models are more applicable as the run time development can be better controlled.

7.1.4 Comparing the run time of the two models

We see that the two models' run time development is the same along the different problem dimensions. However, the SSDP clearly requires more calculations than the PSDP. This is not surprising, as there are fewer restrictions on the inspection decisions in the former. While the PSDP requires a fixed interval for inspections along the entire planning horizon, the SSDP does not. The SSDP's inspection choices is only restricted by the the maximum interval. However, as the run time complexity is the same for both models, they should be applicable for the same problem sizes.

7.2 Resulting policies from the Base Case solutions

In this section, we look at the resulting policies from solving the PSDP and the SSDP. Primarily, we look at non-trivial maintenance and inspection policies caused by dependencies. The policies provided are from the first stage ($n = 1$). The models do not necessarily provide the same policies for all stages, but they are fairly similar, and the policies from any stage would provide a good base for discussion.

7.2.1 Maintenance rate

One of the benefits our models have over conventional condition-based maintenance (CBM) methods is the ability to consider dependencies between components within a system. In this thesis, we focus on the economic dependencies between components. The two costs parameters that influence dependencies in the system are fixed maintenance cost and cost of inspection. Fixed maintenance cost remains the same regardless of the number of maintained components in a stage. Hence it can be beneficial to maintain several components at once. Thus we might end up with different maintenance thresholds (i.e. the state triggering maintenance on a component) when considering components in combination instead of considering them individually. Similarly, optimal inspection intervals for components considered individually might differ from each other. Inspecting at all optimal suggestions for every component individually is unlikely to be the best solution (at least not when inspection costs are significant). Considering the components in combination can suggest an inspection interval that is better when considering the entire system's total costs.

We use the term *maintenance rate* to describe the number of times a component in a given state is maintained relatively to all possible state combination for the other components. In equation 7.1 we define the maintenance rate, R^M_{ikn}:

$$R^M_{ikn} = \frac{1}{(|\mathcal{S}|-1)^{(|\mathcal{K}|-1)}} \sum_{s_n \in \mathcal{S}'} x^*_{kn}(s_{kn}), \qquad n \in \mathcal{N},\ k \in \mathcal{K},\ \mathcal{S}' \subset \mathcal{S}^{\text{sys}} \setminus s_{kn} \neq i, |\mathcal{S}| \tag{7.1}$$

We calculate the maintenance rate for all components and all of their possible states, except the failure state $|\mathcal{S}|$. $\mathcal{S}^{\text{sys}}$ is the set of all possible system states. We have $|\mathcal{S}^{\text{sys}}| = |\mathcal{S}|^{|\mathcal{K}|}$, but as we consider one of the components for each maintenance rate, and do not consider the failure state, we end up with $(|\mathcal{S}|-1)^{(|\mathcal{K}|-1)}$ combinations of system states where $s_{kn} = i$. In short, for all input state vectors $s_n = [s_{1n},\ ...,\ s_{kn},\ ...,\ s_{|K|n}]$ where $s_{kn} = i \in \mathcal{S}$ and $s_n \in S^{sys}$, how often does $x_{kn} = 1$?

For the Base Case with four components, there are $4^3 = 64$ combinations of state vectors when fixing one component's state. If a policy suggests maintaining a component in state 2 for 48 out of the 64 possible combinations, we divide 48 by 64, and the maintenance rate is 0.75.

The maintenance rate provides an intuitive understanding of our models' maintenance policy. When the maintenance rate is 1 for one of the component's states, the associated component should always be maintained when an inspection reveals that the component is in this state. When the maintenance rate is 0 for a state, the component should not be maintained in this state. Using the maintenance rate, we can get a strong indication of which CBM-policies will perform well.

The maintenance rate is also a useful measure when detecting non-trivial decisions suggested by the models, i.e., detecting that a component is in a specific state after inspection sometimes triggers maintenance of that component, but not always. When we consider CBM-policies, the maintenance rate will always be either 0 or 1, meaning that a component will either always or never be fixed for a given state. Generally, whenever the maintenance rate is between 0 and 1, non-trivial maintenance policies will lower the average total cost.

Maintenance rate PSDP

$k \setminus s_{kn}$	1	2	3	4
K0303	0.000	0.000	1.000	1.000
K0305	0.000	1.000	1.000	1.000
K0601	0.000	0.875	0.906	1.000
K1002	0.000	0.000	0.906	0.938

Maintenance rate SSDP

$k \setminus s_{kn}$	1	2	3	4
K0303	0.000	0.000	1.000	1.000
K0305	0.000	0.859	1.000	1.000
K0601	0.000	0.828	0.906	1.000
K1002	0.000	0.000	0.859	0. 938

Figure 7.4: Maintenance rate from the Base Case for the PSDP and the SSDP

Figure 7.4 shows the maintenance rate for the solutions from the PSDP and SSDP, for $n = 1$. The PSDP model provides CBM-like policies for the pole (K0303) and crossbar (K0305), suggesting to always maintain the mast in state 3 and above, and the crossbar in state 2 and above. For the insulator (K0601) and the cable (K1002), the policies are more complex. If we look at the K0601 for both the PSDP's and the SSDP's table, we see that the maintenance rate for state 2 is 0.875 and 0.828 respectively. This suggests that we should maintain the component when detected in state 2 in most cases, but not all.

7.2.2 Complex suggestions from the models

Knowing that some dependencies influence the optimal decision when the state of K0601 in stage 1 is 2 (that is, $s_{K0601,1} = 2$), we investigate this further. Table 7.1 present the optimal decisions for a selection of system states, where variations in the second component's state in stage 1 ($s_{K0305,1}$) significantly influence the maintenance decision of K0601. All components except for K0305 are in state 2. When $s_{K0305,1} = 1$ (first row) both models suggests no maintenance of the second component ($x^*_{K0305,1} = 0$), and also no maintenance for the third ($x^*_{K0601,1} = 0$). But when $s_{K0305,1} = 2$ (second row), the PSDP suggests maintenance for both components ($x^*_{K0305,1} = 1$ and $x^*_{K0601,1} = 1$), while the SSDP continues to suggest no maintenance. However, when $s_{K0305,1} = 1$, the SSDP model also suggests maintenance for both components. We further note that the time until maintenance suggested by the SSDP increases from 3 to 6 years. An interpretation of this is that since the system state is improved, the probability that we would like to maintain over the next few years is so low it does not justify the associated inspection cost for checking the system.

State (s_1)	$x^*_1(s_1)$ PSDP	z^* PSDP	$x^*_1(s_1)$ SSDP	$z^*_1(s_1)$ SSDP
[2, 1, 2, 2]	[0, 0, 0, 0]	9	[0, 0, 0, 0]	3
[2, 2, 2, 2]	[0, 1, 1, 0]	9	[0, 0, 0, 0]	3
[2, 3, 2, 2]	[0, 1, 1, 0]	9	[0, 1, 1, 0]	6

Table 7.1: The table shows optimal policies suggested by the SDPs when used on the Base Case for a selection of different system states in stage 1, where K0305's state varies. For each vector, the indices are associated to K0303, K0305, K0601 and K1002 respectively.

7.2.3 Inspection intervals

The SSDP's dynamic inspection decision property is further highlighted by Table 7.2. Here we vary the state of the fourth component, K1002. In this case, the PSDP and SSDP suggest the same maintenance decisions, but we see some interesting variations in the suggested time until the next inspection from the SSDP. When $s_1 = [1,\ 1,\ 3,\ 2]$ it suggests 6 years until the next maintenance. When the state of the fourth component increases to 3, the SSDP suggests to inspect in 5 years ($s_{K1002,1} = 3$, shown in the second row of Table 7.2). However, when the state of the fourth component (K1002) increases to 4, both the SSDP and PSDP suggest maintaining the components K0601 and K1002. Due to the system's improved state because of maintenance, the SSDP suggest waiting 8 years until conducting another inspection.

As we see in Table 7.2, the SSDP suggests inspecting significantly earlier than the PSDP for some of the possible system states. The optimal inspection interval given by the PSDP is 9 years, while the average suggested inspection interval from the SSDP is 7.496 in the first stage. For the Base Case, we have a total of 256 possible system states ($(|\mathcal{S}| - 1)^{(|\mathcal{K}|)} = 4^4$). Table 7.3 shows the frequency distribution of suggested inspection intervals of the SSDP. For most system states, the model suggests

State (s_1)	$x_1^*(s_1)$ PSDP	z^* PSDP	$x_1^*(s_1)$ SSDP	$z_1^*(s_1)$ SSDP
[1, 1, 3, 2]	[0, 0, 0, 0]	9	[0, 0, 0, 0]	6
[1, 1, 3, 3]	[0, 0, 0, 0]	9	[0, 0, 0, 0]	5
[1, 1, 3, 4]	[0, 0, 1, 1]	9	[0, 0, 1, 1]	8

Table 7.2: The table shows optimal policies suggested by the SDPs when used on the Base Case for a selection of different system states in stage 1, where K1002's state varies. For each vector, the indices are associated to K0303, K0305, K0601 and K1002, respectively.

inspecting with 8-year intervals. However, it will suggest shorter intervals in some situations, e.g. for the first two combinations in Table 7.2.

Years until next inspection	1	2	3	4	5	6	7	8	9	10
Frequency in stage 1	0	0	9	5	7	52	3	180	0	0
Frequency in stage 15	0	0	9	5	7	52	3	180	0	0
Frequency in stage 30	0	0	9	5	7	50	4	181	0	0

Table 7.3: Distribution of suggested inspection intervals by the SSDP for Base Case

From the maintenance rates in Figure 7.4 (in Section 7.2.1), we also note that the maintenance policy for the PSDP is more conservative than the policy suggested by the SSDP. The PSDP has a higher average maintenance rate and suggests maintenance for more system state combinations than the SSDP. For example, the PSDP suggests maintenance for all system states where K0305 is in state 2 in stage 1 ($s_{K0305,1} = 2$). The SSDP only suggests maintenance in 85.9% of the system states where $s_{K0305,1} = 2$. This highlights one of the most important differences between the PSDP and SSDP: As the SSDP allows more flexibility, it chooses a less conservative maintenance policy but compensates with more frequent inspections. This also emphasizes the interconnection of maintenance and inspection.

It is worth to note that the SSDP suggest very similar policies in stages in the first part of the planning horizon, but will have larger variations towards the end. Table 7.4 shows the frequency distribution of z_n^* for some of the last 10 stages.

Years until next inspection	1	2	3	4	5	6	7	8	9	10
Frequency in stage 40	0	0	5	7	9	52	183	0	0	0
Frequency in stage 42	0	0	7	7	58	48	0	0	136	0
Frequency in stage 44	0	1	10	58	5	0	182	0	0	0
Frequency in stage 46	0	0	13	5	238	0	0	0	0	0
Frequency in stage 48	0	4	252	0	0	0	0	0	0	0

Table 7.4: Distribution of suggested inspection intervals by the SSDP for Base Case

Here we see significant and somewhat unpredictable variations. This is the consequence of the SSDP trying to adjust the inspection intervals to minimize the expected end of horizon cost. This adjustment stresses the importance of setting a reasonable end of horizon cost. Alternatively, one can run the model for longer horizons and use only the first part, as the suggested policies will approach a steady-state for stages far from the end of the planning horizon.

7.3 Comparing the performance of different policies on the Base Case

In this section, we compare different policies. We apply the policies returned from the Periodic and Sequential SDPs on the Base Case, and compare their performances to conventional policies. The conventional policies are summarized in Section 6.4 of the Case Study. We consider the "run to failure" policy (RTF), along with several variations of the condition-based maintenance-policy (CBM-policy) type. In Section 7.3.1, we derive another "conventional" policy from our model-results. We then see how total costs are affected when simulating with policies in Section 7.3.2. Section 7.3.3 examines how often the system fails with the different policies before investigating how a different initial state of the system's components affects some of the policies with respect to the total cost. In Section 7.3.5, we look at how the SSDP exploits inspection decisions' flexibility.

7.3.1 Deriving a CBM from our models

When comparing our models' results, we also take a simple approach to derive an "optimal" CBM based on the resulting policies from the SDPs. In this CBM, we let the inspection interval equal the optimal interval returned by the Periodic SDP, and base the maintenance levels on the component-states that both the PSDP and the SSDP most frequently maintain at, in their optimal policies. We refer to this policy as the *CBM-optimal* (CBM-O) as it is derived from optimal policies.

The reader should note that the "conventional" CBM-O policy is not derived based on intuition, but rather the data at hand and adapted to a widely used format. Furthermore, the difference between the PSDP and the CBM-O is subtle but distinct. While the CBM-O has individual thresholds for the components, the maintenance decisions returned by the PSDP are based on the entire system's state, meaning that an individual component may be maintained based on the state of the other components in the system. The CBM-O has a static threshold for all components, and will thus *always* yield the same choice for a specific component-state.

We provide the CBM-O-policy in the bottom row of Table 7.5, where we have also reproduced the other conventional policies as they are used on all simulations in this section. For the CBM-O-policy, we use the optimal periodic interval returned from the PSDP, which is 9. Both models also return

Policy	Interval	K0303	K0305	K0601	K1002
RTF	10	-	-	-	-
CBM-S-H	2	4	4	4	4
CBM-B	5	3	3	3	3
CBM-E-L	8	2	2	2	2
CBM-O	9	3	2	2	3

Table 7.5: The table shows the inspection interval and state-thresholds of the conventional policies, including the derived CBM-O, for the components used in the Base Case.

varying maintenance levels for each component depending on the other states of the system. The maintenance thresholds of the CBM-O are the lowest state where the SDPs yield a maintenance rate higher than 0.5 in stage 1.

7.3.2 Total cost from simulations

Table 7.6 gives the results from running 10 000 simulations of the Base Case's entire planning horizon. The actual total costs of failures, inspections and maintenances are calculated for each policy in all simulations, and the average is returned, along with the maximum total cost, the minimum total cost and the standard deviation.

We see that the Periodic SDP and the Sequential SDP obtain the lowest average cost, which stems from the fact that these policies consider the economic dependencies between components. The PSDP and SSDP respectively achieve 2.2% - 11.9% and 2.6% - 12.2% lower average cost than all the conventional policies suggested in Chapter 6. The SSDP has a lower total cost than the PSDP, as the decision maker can decide both on maintenance (in a stage with inspection) and when to carry out the next inspection. The flexibility of carrying out or postponing an inspection based on the system's current state makes the information from the current inspection more valuable for the SSDP than the PSDP. However, the decision maker following the SDP-policy can make maintenance decisions based on the states of all components, as opposed to the CBM-policies, which consider each component individually during an inspection.

The run to failure-policy (RTF-policy) obtains a low minimum total cost from the simulations. The minimum cost-scenario is a simulation with very little deterioration on the system throughout the 50 years. For the RTF-policy, inspections are done only when required, every tenth year. With a planning horizon of 50 years and a required inspection in year 1, the RTF policy will have five inspections (years 1, 11, 21, 31 and 41) resulting in a cost of 10. This is the lowest possible cost of a 50-year period, as a system technically may not deteriorate over the 50 years at all (although this is highly

Policy	Avg.	Max.	Min.	Std. Dev.
RTF	240.04	518.86	22.42	68.34
CBM-S-H	253.11	584.88	62.42	64.79
CBM-B	228.14	508.88	30.32	66.02
CBM-E-L	231.22	511.30	37.14	67.80
CBM-O	**225.92**	**504.70**	**29.14**	**66.93**
PSDP	**223.04**	**512.88**	**22.32**	**66.97**
SSDP	**222.31**	**523.38**	**24.32**	**66.63**

Table 7.6: The table shows costs in kNOK from 10 000 simulations for the different policies, with the derived policies' performances in bold.

unlikely). As CENS is 15, the minimum-cost simulation from the RTF-policy can not include a system failure, as the total cost then must exceed 25 (10 + 15 + replacement cost). The total cost of 22.42 comes from five inspections and the end of horizon cost, implying that the system deteriorated some during 50 years, but not enough to fail. The minimum cost for the PSDP has probably occurred in the same simulation, where maintenance resulted in even lower total cost (as the end of horizon cost seems to have lowered more than the cost incurred of maintaining the system). From the SSDP minimum cost, it is likely that the policy was similar to the PSDP's, but with an extra inspection, resulting in a higher cost than both the RTF and PSDP.

Compared to the RTF, the other policies have a lower level of risk related to average cost as they inspect the system more often, leading to a "best-case scenario" with a higher minimum total cost for all other policies than the PSDP. As the minimum cost-scenarios are very unlikely, and it is difficult to predict which policy will return the lowest cost from a single simulation, the risk level might be better to derive from the standard deviation. We see that the RTF-policy varies the most from the average total cost in the simulations. The reader should note, however, that the RTF does not yield the highest total cost. This is because both informed and reasonable decisions may still turn out to be the worst possible choice. For instance, if inspections are carried out often, and the system is maintained at low state levels, there is still an off-chance that the system fails in the same stage as maintenance and inspection happen. Over a large number of simulations, this may happen several times within a single simulation. This seems to impact the SSDP-policy to some degree, and especially the CBM-short-high-policy (CBM-S-H-policy), as the latter has the highest total cost from any simulation. This is reasonable, as the policy both inspect quite frequently and maintain at high state-values.

The CBM-S-H-policy also has the highest average total cost, implying that the inspections are too frequent and the maintenances are too rare. This policy's standard deviation implies a trade-off between risk and reward, where we see that CBM-S-H returns the lowest standard deviation of all policies. This is partly due to this policy's minimum cost, as it inspects every other year. The expected total cost is consequently very high. The same argumentation applies to the CBM-extended-low-policy (CBM-E-L-policy), a seemingly less conservative policy that achieves much lower average total cost, but a higher standard deviation. This again implies the trade-off between higher risk and lower expected total cost. Nevertheless, we see that the CBM-balanced-policy (CBM-B-policy) achieves both lower average cost and standard deviation than the CBM-E-L and that the RTF has even higher average cost and standard deviation. This observation implies that some decisions are beneficial both for risk and expected cost.

The CBM-O-policy also provide impressive results, with 1.0% - 10.7% lower average cost than all the other conventional policies. Although this policy does not benefit from the flexibility of the inspection decisions like the SSDP or maintenance decisions like the SSDP and PSDP, the CBM-O outperforms all the other conventional policies with respect to average cost. When solving a case to optimality with the SDPs, deriving the CBM-O is trivial.

7.3.3 System failure rates

Another measure of risk is the number of failures that occur during a 50-year-simulation. No policies can guarantee that a system won't fail as a system may fail in the same year that all its components began as "good as new". This modelling choice incorporates the possibilities of sudden shocks such as lightning, or intense weather discussed in Chapter 2. However, over several 50-year simulations, the expected failure rate will differ based on the applied policy.

Table 7.7 shows the average system failures over 10 000 simulations for all policies. That is, the average number of times a single component deteriorates to the failed state, forcing failure costs and a replacement. We see that the RTF is a strategy that has a higher risk of failure, while the policy with the highest average cost, the CBM-S-H, has the lowest average failures.

The risk discussed concerning the standard deviation of total costs from all policies seems to correspond with the average failures. However, we see that the CBM-O has more failures than the CBM-E-L, although it had a lower standard deviation. This implies that other factors than the risk of failure affects the standard deviation of total costs, such as the minimum total cost.

We also see that the SSDP has fewer failures than the PSDP. This may result from the SSDP's ability to adapt to outcomes with both maintenance and inspection decisions. It is also interesting to note that the CBM-O has fewer failures than the PSDP and a higher failure rate than the SSDP. The latter is because of how we derive the CBM-O; we considered the maintenance rates of the SDP-policies

Policy	Avg. System Failures
RTF	7.35
CBM-S-H	4.97
CBM-B	5.10
CBM-E-L	5.31
CBM-O	**5.58**
PSDP	**5.73**
SSDP	**5.46**

Table 7.7: The table shows system failure rates from 10 000 simulations on all policies, with the derived policies' performances in bold.

and set thresholds for component repairs where the policies returned from the two SDP-models "only" would maintain in over than 50% of the possible system states. This makes the CBM-O more conservative than the PSDP as they have the same inspection interval, while the adaptability of inspections that the SSDP benefit from makes it the least prone to failure of these three policies. The conservative trait of the CBM-O can also be seen by it's significantly higher minimum total cost from Table 7.6, compared to the SDP-policies.

In Figure 7.5, we see the frequency distribution of failures per 50-year period for the PSDP, the SSDP, the RTF and the CBM-B. We omit some of the conventional policies to make the figure more readable. The curves for each policy is distributed around the average system failures in Table 7.7 and also gives an impression of the standard deviation of failures. This standard deviation is different from the standard deviation of total costs, and may again better represent the risk related to a power outage. We see that the CBM-B-policy is denser than the others, which is expected as a more conservative policy will lead to fewer failures in simulations with more deterioration on the system.

The Periodic SDP also seem to have a denser distribution than the Sequential SDP. The adaptability of the SSDP may explain this. In some simulations, the system may deteriorate more than expected, but not enough to trigger maintenance. This situation can arise when only a single component has deteriorated to a state where it is only repaired if other system components have also adequately deteriorated. If the rest of the system is in a state of little deterioration, the fixed cost of maintenance makes it more economical to leave the more deteriorated component in its current state. In these situations, the SSDP might recommend a shorter inspection interval, while the PSDP is bounded by the fixed inspection interval of 9 years. If this is the case, the SSDP may avoid some failures, happening with the PSDP, giving the curves we see in Figure 7.5.

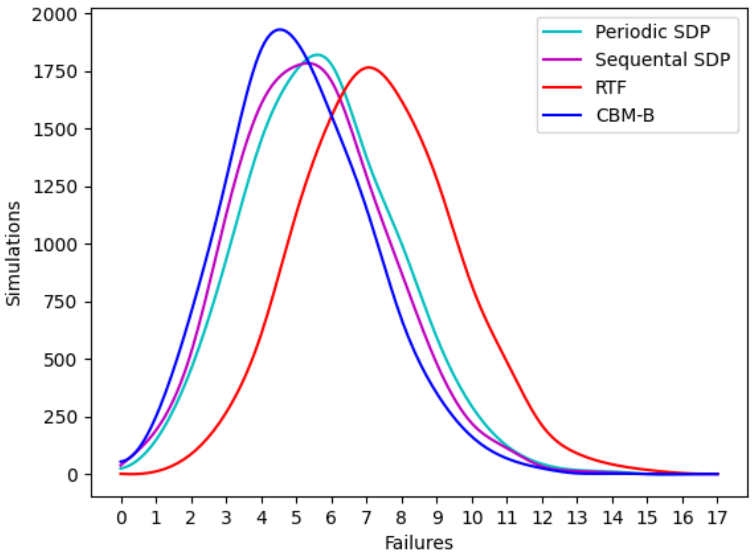


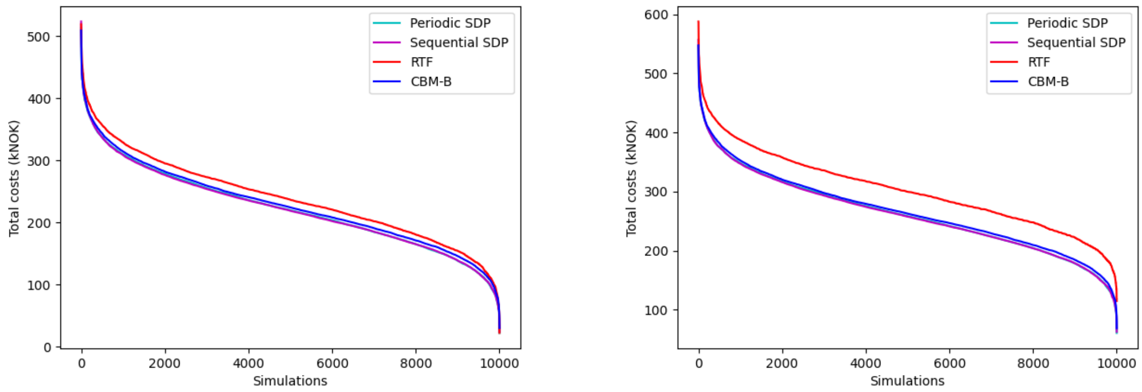
Figure 7.5: The figure shows the frequency distribution of failures per simulation for four of the policies, over 10 000 simulations.

7.3.4 Simulating over a different initial system state

The resulting costs and failures from simulations are naturally affected by the problem case in question and by the assumption of a brand new system in year 1. The SSDP return optimal policies for all possible initial states, allowing us to choose optimal policies for the Base Case also when we assume a different initial state for the system. However, for the PSDP, the optimal inspection interval is derived for the initial state where all components are in state 1.

In Figure 7.6, we simulate on two different initial states: One where all components are in state 1 in year 1 (Figure 7.6a), and one where all components are in state 3 in year 1 (Figure 7.6b). An initial state where all components are in state 3 should yield a close to optimal PSDP, as the policy will fix all components in the first year. Again, we choose only four of the policies for better readability. The total costs from all simulations are plotted for the RTF, the CBM-B and the two SDPs, where the costs are in decreasing order. We see that the RTF is increasingly worse than the other policies regarding total costs when the initial state is worse. This highlights the impact of the RTF- policy. When the system is quite deteriorated in year 1, the average number of failures across the 50-year period will naturally increase with no maintenance on the system. The difference between the RTF and the other policies is more subtle when the system starts in a "good as new"-state.

Furthermore, we see that the SSDP, the PSDP and the CBM-B achieve similar total cost with a worse initial state, with the SSDP being somewhat better than the PSDP, which again outperforms the CBM-



(a) Simulations when all components begin in state 1. **(b)** Simulations when all components begin in state 3.

Figure 7.6: The figure shows the total costs of 10 000 simulations in decreasing order for four of the policies when all components in year 1 have state 3 and 1.

B. This should not be surprising considering the simulation results on a "good as new"-system. Both the SDP-policies and the CBM-B-policy will repair the system in the initial state which is less costly than bearing the risk of system failure from all the components in state 3, which is the choice when following an RTF-policy. Thus, the cost of repairing the whole system in year 1 is the difference between the two graphs for the SDPs.

7.3.5 The flexibility of the Sequential SDP

In this section, we discuss the ability of the SSDP to adapt to system deterioration both with the maintenance decisions and inspection decisions. The latter ability differs from all the other policies and has proven advantageous concerning total cost. Furthermore, it achieves fewer failures than the PSDP.

Figure 7.7 shows the different inspection intervals chosen from the SSDP over 10 000 simulations. The forced inspection in year 1 is not considered a chosen interval and omitted. To avoid any misunderstandings from the previous discussion, we emphasize that the system's initial state has all components in state 1. We note that the most frequent inspection interval is 8 years, but that all possible intervals between 2 years (barley visible in the figure, happening 27 times) and 9 years occurred in the simulations. Additionally, more often than not, the inspection interval is *different* from 8 years, showing that the optimal inspection interval varies quite a lot in the simulations.

As the initial state is 1 for all components, the resulting first inspection interval will be equal for all the simulations. For the Base Case, this interval is 8, which dramatically affects this interval's frequency as 10 000 choices come from the exact same starting position. We, therefore, see that after the first

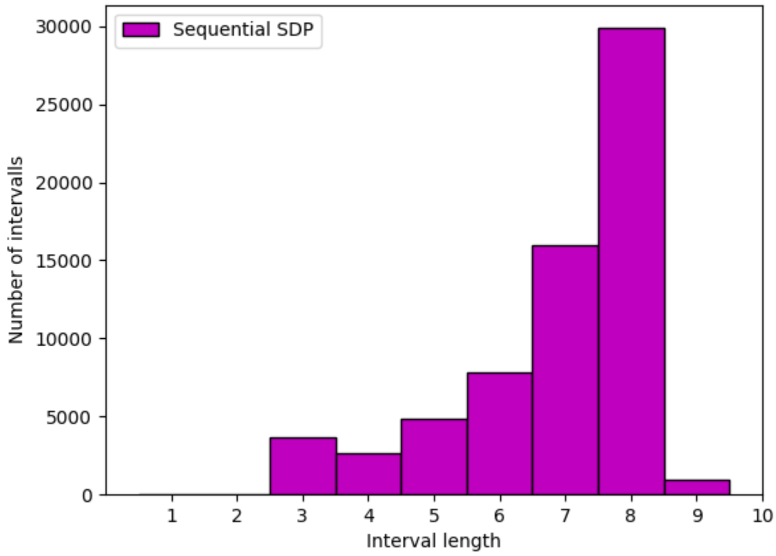


Figure 7.7: The figure shows the frequency distribution of different inspection intervals chosen by the Sequential SDP over 10 000 simulations.

year, the distribution is more even. Although 8 years is still the most frequent interval, 7 years occurrs almost as often after the first inspection choice. The results discussed in this section show that the flexibility of inspection decisions makes the SSDP perform better than the other policies with respect to the total cost.

For the Base Case, we have also seen that the SSDP has similar results to especially the PSDP and the CBM-O. Although the latter two are outperformed by the former, it is interesting to see that even a CBM-policy can be tuned to achieve close to optimal results. The reader should note that deriving the CBM-O required solving both models for the case and that its result may vary based on the case's parameters. Nevertheless, searching for a good condition-based maintenance-policy for a larger system may be aided by solving smaller cases with less complex systems to approximate the optimal state-threshold for each component.

7.4 Varying case input parameters

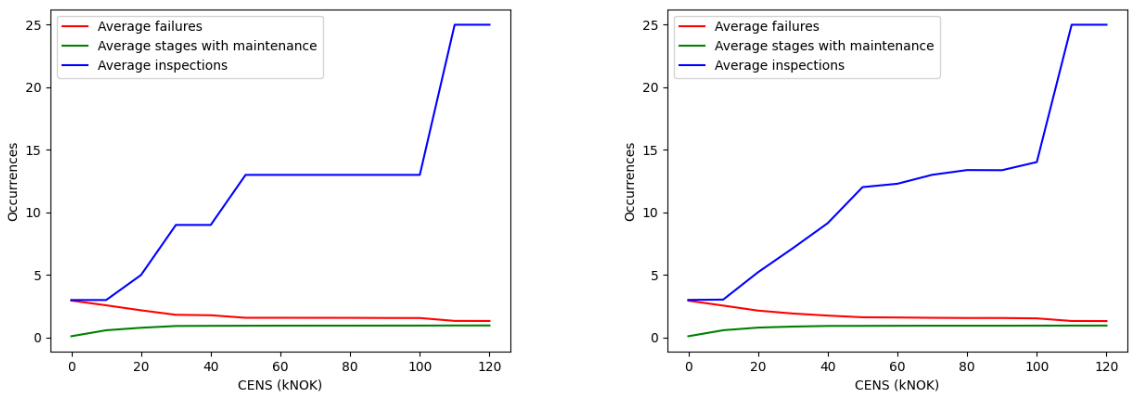
The Base Case only considers one out of many possible combinations of input parameters. As the resulting policies from our models are dependent on the case parameters, it is relevant to explore varying them influence the solutions from the SDPs.

In Section 7.4.1 we explore how changes in the penalty cost of failure influence the relative costs from, and occurrence of inspections, maintenance and failures. In sections 7.4.2 and 7.4.3 we do similar analyses for variations in inspection cost and fixed maintenance cost, respectively.

7.4.1 Varying failure cost

Both the PSDP and the SSDP seek to adjust the number of inspections, maintenances and consequently expected failures such that the total expected costs are minimized. Hence, the associated costs from these events play a significant role in the policies the models suggests. For example, a power grid line with high CENS, but low inspection and maintenance costs will be inspected and maintained more frequently than a remote line with lower CENS, but higher inspection and maintenance costs.

The graphs in Figure 7.8 shows how often inspections, maintenance and failures occur on average over 1 000 simulations when varying CENS for both models. We notice that increasing CENS leads to an increase in the number of inspections and maintenances, while the number of failures decrease.



(a) PSDP **(b)** SSDP

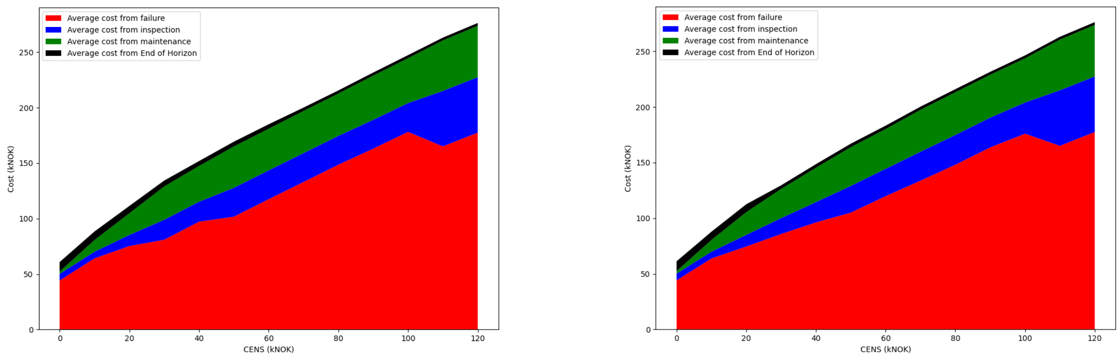
Figure 7.8: The figure shows the number of occurrences of failures, maintenances in a stage and inspection when varying CENS for the Sequential and Periodic SDPs. The used cases are VCE_01 - VCE_13.

The change in the number of maintenances is most significant for the smaller CENS-values. The average number of maintenances is 0.106 and 0.111 when CENS is 0 for the PSDP and SSDP, respectively. When CENS is 10 kNOK, it changes to 0.547 and 0.566 and then to 0.786 and 0.800 for a CENS of 20 kNOK. For the rest of the interval, the increase is significantly lower, with the average number of maintenances being 0.970 for both models when CENS is at 120 kNOK.

We see that the number of failures becomes less frequent as the cost of a failure increases. The average number of failures is 3.057 for both models when CENS is 0. When CENS is 50 kNOK, the failure rate is 1.688 for the PSDP and 1.699 for the SSDP. Then the failure rate remains very stable up to a CENS of 100 kNOK, being 1.650 and 1.621, but makes a jump down to 1.405 for both models when CENS is 110. This change coincides with the change in inspections at the same CENS value. We also see a relatively small, but significant, jump in the number of maintenances at the same time from 0.951 to 0.969 for the PSDP and 0.954 to 0.969 for the SSDP.

Considering the number of inspections when varying CENS, we make several interesting observations. First, we notice that the graph for the SSDP is smoother than that of the PSDP. This is because the PSDP defines fixed inspection intervals that do not change throughout the planning horizon, resulting in the same number of inspections for all simulations for each case. In comparison, the SSDP is more flexible regarding inspections and can adjust the inspection interval to the situation. Therefore, it will not get the same discrete jumps in the number of inspections as the PSDP.

The number of inspections must be seen in relation to the length of the planning horizon. For these cases, we consider a planning horizon of 25 years. With a maximum allowed inspection interval of 10, we must have at least 3 inspections. We can at most have 25 inspections, meaning inspection every year. In the plots in Figure 7.8, where CENS vary from 0 to 120 kNOK, the number of inspection range from the lower limit of 3 to the upper limit of 25 inspections for both models.



(a) PSDP **(b)** SSDP

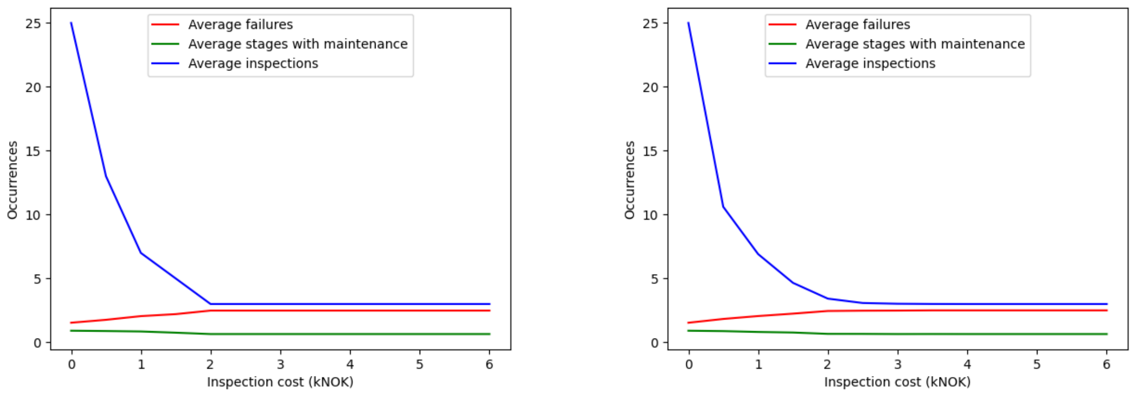
Figure 7.9: The figure shows a change in costs when varying CENS for the Sequential and Periodic SDPs. The used cases are VCE_01 - VCE_13.

From Figure 7.9, we see the costs' consequential changes when varying CENS. When CENS changes from 100 kNOK to 110 kNOK we can recognize similar graph changes for the costs, as shown in Figure 7.8. It is interesting to note that the increase in inspection cost is much more significant than the increase in maintenance cost. This implies that in this specific situation when CENS changes from 100 kNOK to 110 kNOK, the models primarily use inspections as a mean to mitigate failure. It changes the inspection interval from 2 years to 1 year for the PSDP and the average inspection interval from 1.992 to 1.0 for the SSDP. Hence, the costs from inspection almost doubles (92% increase for the PSDP and 79% for the SSDP) while the increase in maintenance cost is 12.4% and 13.4% for the PSDP and the SSDP, respectively. A possible alternative policy could have been to set a more conservative maintenance policy but not increasing inspection frequency.

One might intuitively expect that the cost from failures should be 0 when there is no penalty cost from failures (CENS = 0). We see from Figure 7.9 that this is not the case. The cost of failures has the most significant contribution to the total cost. The reason is that a failure still incurs the costs from repairing the failed component. Further, as the cost of failure is similar to the cost of maintenance, it will in many situations make sense to use a "run to failure"-policy. However, we see that the models do suggest maintenance in some situations where CENS is 0, as the cost from maintenance is nonzero. The reason is that several components can be maintained at once, sharing the fixed maintenance cost and making it beneficial to maintain rather than applying a pure "run to failure"-policy.

7.4.2 Varying inspection cost

This section will explore how variations in the inspection cost influence the solutions from our models in a similar fashion as the previous section 7.4.1.

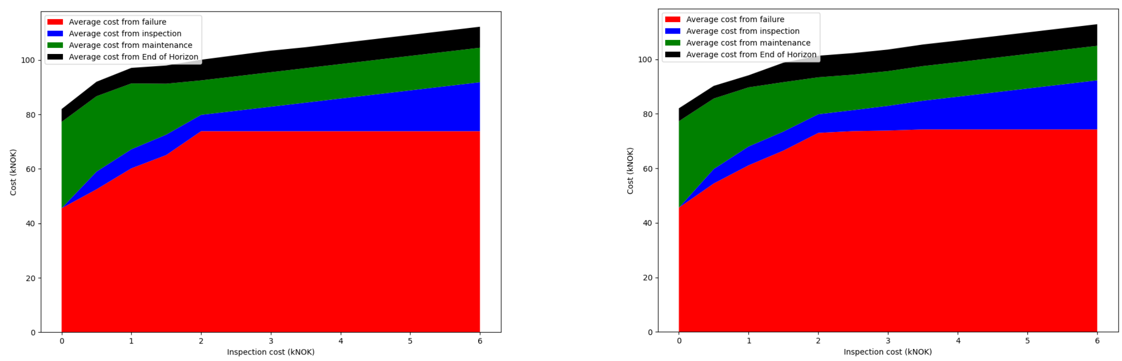


(a) PSDP **(b)** SSDP

Figure 7.10: The figure shows the number of occurrences of failures, maintenances in a stage and inspection when varying inspection cost for the Sequential and Periodic SDPs. The used cases are VCI_01 - VCI_13.

Studying Figure 7.10 we see that the number of inspections also vary between the maximum and the minimum allowed number of inspections when the inspection cost vary, however oppositely correlated. Not surprisingly, the models suggest inspecting every year when inspection incur no cost. As the cost of inspection increase, the frequency of inspections decrease. Fewer inspections reduce the power grid operator's ability to carry out well-timed maintenances. This leads to a reduced number of maintenances, which leads to more failures. The interaction between inspections and maintenances is further underlined when studying the development of costs with varying inspection cost, shown in Figure 7.11.

With the first increments of increased inspection cost, the total cost from inspections remains relatively stable as each inspection's cost increases, but the frequency decreases. Meanwhile, the average total cost from maintenance decreases, causing both the average cost from failure and the end of horizon cost to increase.



(a) PSDP **(b)** SSDP

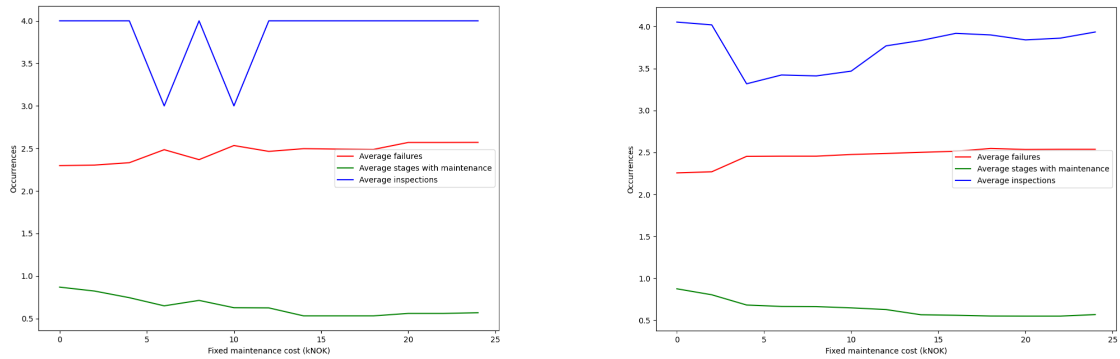
Figure 7.11: The figure shows a change in costs when varying inspection cost for the Sequential and Periodic SDPs. The used cases are VCI_01 - VCI_13.

When the inspection cost is 2 kNOK or more, the inspection rate stabilizes at the maximum inspection interval of 10 years (exactly 10 years for the PSDP, and an average close to 10 years for the SSDP). Consequently, the costs also stabilizes, where the total cost from inspections is the only one increasing due to its increase in cost per inspection.

7.4.3 Varying fixed maintenance cost

In this section, we study the implications on the models' solutions when varying the fixed cost of maintenance. From Figure 7.12, we see that there are significantly less change in the number of inspections when varying the fixed maintenance cost, compared to the change in the number of inspections when varying CENS and inspection cost. This might indicate that the correlation between fixed maintenance cost and the ideal number of inspections is relatively small.

We do however notice some variation. For a fixed maintenance cost of 6 and 10 kNOK, using the PSDP policy results in 3 inspections, compared to 4 inspections for the cases with other fixed maintenance costs. These results also suggest that less frequent inspection results in fewer maintenances and more failures, seeing the correlating drop in maintenance and inspection rate, but increase in failure rate at $C^{MF} = 6$ kNOK and $C^{MF} = 10$ kNOK.

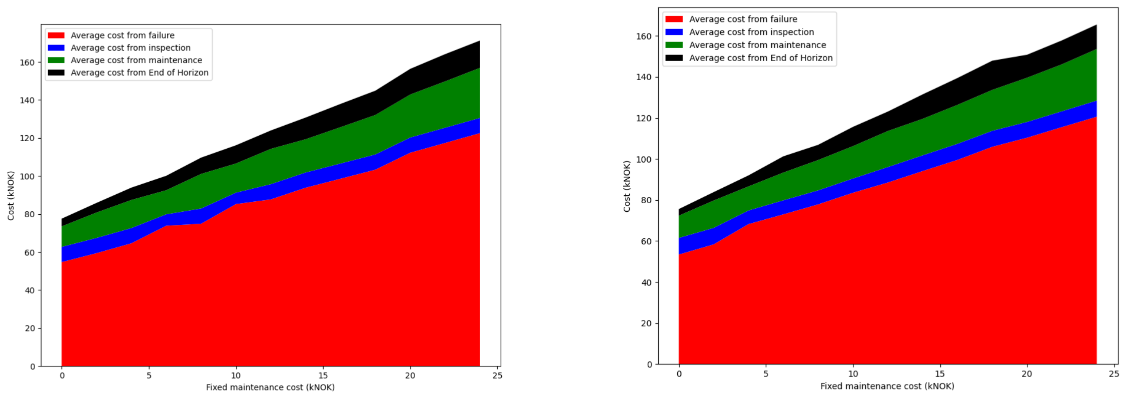


(a) PSDP **(b)** SSDP

Figure 7.12: The figure shows the number of occurrences of failures, maintenances in a stage and inspection when varying the fixed maintenance cost for the Sequential and Periodic SDPs. The used cases are VCM_01 - VCM_13.

Figure 7.12 also shows that the number of stages with maintenance is significantly higher when there is no fixed maintenance cost. With no fixed maintenance cost, our models do not have any incentives to coordinate different components' maintenance. Thus maintenance takes place in more stages. When the fixed maintenance cost increase, the models seeks to reduce the expected total costs by maintaining several components at once, thus sharing the fixed maintenance cost.

Figure 7.13 shows the average of the different contributions to the total cost when varying the fixed maintenance cost. Interestingly, the average total cost from maintenance is lower for $C^{MF} = 6$ kNOK compared to $C^{MF} = 4$ kNOK (12.67 kNOK vs 14.79 kNOK), when using the policy from the PSDP. This is not trivial, as the cost from maintenance typically increases as the fixed maintenance cost increases, as is the case for all other values in the graph.



(a) PSDP **(b)** SSDP

Figure 7.13: The figure shows a change in costs when varying the fixed maintenance cost for the Sequential and Periodic SDPs. The used cases are VCM_01 - VCM_13.

We see the same behaviour from the policy suggested by the SSDP, but here at $C^{MF} = 4$ kNOK, being 11.74 kNOK compared to 13.38 kNOK at $C^{MF} = 2$ kNOK. For both these instances, we see a large increase in the average cost from failure. An interpretation is that for some system states, maintenance becomes too costly for the models. Thus they instead take on a more considerable risk of failure.

7.5 2K-SH: The Two-Component Sequential Heuristic

While our models solve a system with a few components to optimally, their run time make practical use for operational purposes unrealistic. Thus, other methods are needed to obtain useful policies. This section presents a heuristic, named 2K-SH, that uses the Sequential SDP to solve several smaller problems to optimality and then combine the solutions to provide an inspection and maintenance policy for a larger system. We provide an algorithm for the procedure of obtaining policies and discuss the heuristic's run time. We also discuss the policies suggested by the 2K-SH before comparing the heuristic's results with the other policies considered in this chapter.

7.5.1 Algorithm

Algorithm 1 presents the pseudo-code for the 2K-SH. We use the notation presented in Chapter 5. To recapitulate, we use C^F for CENS, C^{MF} for the fixed cost of maintenance, C^I for inspection cost and C_{ik}^{End} for the end of horizon cost. z^{max} is the maximum allowed inspection interval and $\mathcal{N}$ is the set of all stages. $\mathcal{K}$ is the set of all components, each component having an individual variable maintenance cost and deterioration matrix. Finally, we have $\mathcal{S}^{sys}$, being all possible system states that s_n can become.

The heuristic uses the same parameters as the main problem when solving the smaller problems using the SSDP, except for inspection cost. We scale this cost with the number of components used to solve the smaller problem, 2, divided by the number of components in the main problem, ($|\mathcal{K}|$). This scaling represents the part of the inspection cost, shared by the components in the smaller problem-instance. This assumption is reasonable, as an inspection provides information about all components. The fixed maintenance cost is kept the same, as the heuristic should not trigger individual components' maintenance frequently. The heuristic provides a policy with maintenance decisions x_n^* and inspection intervals z_n^* for all possible system state s_n in all stages $n \in \mathcal{N}$.

The heuristic generates a solution by combining the SSDP solutions from all possible component pairs (2K) out of the problem's components. When we solve the heuristic for the Base Case with four components, there are six such component pairs (4 choose 2), each component being part of 3 such pairs. If any of the smaller problem solutions suggest maintenance for a component in a given state, that is ${x^*_{kn}}^{2K}(s_{kn}) = 1$ for any of the 2K solutions, the heuristic decides to maintain that component as well ($x^*_{kn}(s_{kn}) = 1$). The inspection interval is found by averaging the optimal intervals form the smaller solutions and rounding them to the nearest integer.

Algorithm 1: Two-component Sequential Heuristic (2K-H)

```
Data: C^F, C^{MF}, C^I, C^{End}_{ik}, z^{max}, 𝒩, 𝒦, 𝒮^{sys}
Result: Policy with x*_n(s_n) and z*_n(s_n) for all system states, s_n ∈ 𝒮^{sys}
components ←list of components k ∈ 𝒦 with maintenance costs and deterioration matrices
temp_solutions ← empty array
for i ∈ {1, ..., |𝒦| − 1} do
    for j ∈ {i, ..., |𝒦|} do
        temp_case ←C^F, C^{MF}, (2/|𝒦|) C^I components[i], components[j]
        temp_solutions ← solve SSDP(temp_case)
    end
end
for n ∈ 𝒩 do
    for s_n ∈ 𝒮^{sys} do
        for k ∈ 𝒦 do
            if x*_{kn}^{2K} is 1 for any solution in temp_solutions then
                x*_{kn} = 1
            else
                x*_{kn} = 0
            end
        end
        x*_n ← x*_{kn}
        z*_n ← rounded average of all z*_n from all temp_solutions
    end
    policy ← x*_n, z*_n
end
return policy
```

7.5.2 Run time of the 2K-SH

We see from the algorithm that when the 2K-SH solves a problem for a system with $|\mathcal{K}|$ components, it solves all possible two-component combination of these $\mathcal{K}$ components to optimality. This means that the heuristic uses the solution procedure of the SSDP $\binom{|\mathcal{K}|}{2}$ times. As $|\mathcal{K}|$ increases, the solution procedure will still solve problem-instances with two components but an increasing amount of these instances. Generally, the number of instances will increase with a complexity of $\mathcal{O}(|\mathcal{K}|^2)$. It is this number that drives the run time of the 2K-SH. The run time complexity of the 2K-SH is considerably more bounded than the run time complexity of the SSDP when increasing the number of components.

Using the 2K-SH on the Base Case with four components means solving the Base Case with two components $\binom{4}{2} = 6$ times. The resulting run time is approximately 75 seconds compared to the nine and a half hours needed to solve the SSDP for four components, which is a vast improvement.

The drawback of the heuristic is that it only considers the dependencies between components in pairs. The reader should note that increasing the number of components in each sub-set solved by the SSDP in the heuristic, still will mean an exponential increase in run time. When solving for instances of $|\mathcal{K}'|$ components where $\mathcal{K}' \subset \mathcal{K}$, the number of combinations that needs to be solved increases with $\mathcal{O}(|\mathcal{K}|^{|\mathcal{K}'|})$ (Cormen et al., 2009). Furthermore, each instance's run time will also increase exponentially when increasing $|\mathcal{K}'|$. However, the proposed heuristic is based on the intuition that sub-sets of all the components in a system may be sufficient to discover the most important dependencies for all components of a larger system. This means that the heuristic's general approach should allow modelling of larger systems, as the size of $\mathcal{K}'$ can be limited.

7.5.3 Policies from the 2K-SH

In Section 7.2.1 we introduce the measure maintenance rate and use it to gain an intuitive understanding of a model. Figure 7.14 shows the maintenance rates from SSDP and our heuristic.

Maintenance rate SSDP

$k \setminus s_{kn}$	1	2	3	4
K0303	0.000	0.000	1.000	1.000
K0305	0.000	0.859	1.000	1.000
K0601	0.000	0.828	0.906	1.000
K1002	0.000	0.000	0.859	0. 938

Maintenance rate 2K-H

$k \setminus s_{kn}$	1	2	3	4
K0303	0.000	0.000	1.000	1.000
K0305	0.000	0.969	1.000	1.000
K0601	0.000	0.906	0.906	1.000
K1002	0.000	0.000	0.906	0.984

Figure 7.14: Maintenance rates from the Base Case for the SSDP and the heuristic

The maintenance rates that the policy suggested by the heuristic share similar characteristics with the policy suggested by the SSDP. We see that the 2K-SH has a somewhat higher average maintenance rate, indicating fewer non-trivial maintenance decisions. This should not be surprising, as the heuristic is unable to capture dependencies of more than two components at the time, i.e. when it is the difference in the third components state that triggers a maintenance decision from (as we saw in Table 7.1 in Section 7.2.2).

The 2K-SH suggest somewhat longer inspection intervals than the SSDP, with the average number of inspections over 10 000 simulations being 7.09 and 7.58 respectively. Deciding on longer inspection intervals is reasonable, as the 2K-HS adopt a slightly more conservative maintenance policy. Table 7.8 shows the distribution of suggested inspection intervals for the SSDP and the heuristic, where we also see that the heuristic has less variation in the inspection interval length it suggests.

Years until next inspection	1	2	3	4	5	6	7	8	9	10
Frequency in stage 1, SSDP	0	0	9	5	7	52	3	180	0	0
Frequency in stage 1, 2K-SH	0	0	0	0	0	24	42	63	127	0

Table 7.8: Distribution of suggested inspection intervals by the SSDP and the 2K-HS for Base Case

7.5.4 Comparing results from different policies with 2K-SH

To benchmark the heuristic's performance, we apply it to suggest a policy for the Base Case and compare those results to the results from other policies we discuss in Section 7.3. Table 7.9 shows the performance of the Sequential SDP, the Periodic SDP and the derived CBM-O compared to the heuristic over 10 000 simulations.

Policy	Avg.	Max.	Min.	Std. Dev.
CBM-O	225.92	504.70	29.14	66.93
PSDP	223.04	512.88	22.32	66.97
SSDP	222.31	523.38	24.32	66.63
2K-H	**223.12**	**523.38**	**26.42**	**66.63**

Table 7.9: The table shows costs in kNOK from 10 000 simulations for the different policies, with the heuristic policy's performance in bold.

We see that the 2K-SH's performance is close to those of the PSDP and the SSDP. The average cost achieved by the heuristic is almost identical to the PSDP, being only 0.4‰higher. Compared to the SSDP, the average cost is 0.4% higher. Furthermore, the performance is significantly better than the CBM-O, derived from the optimal policies of the SDPs. Consequently, the 2K-SH achieves 2.2% - 11.8% lower average cost than all the other conventional policies suggested in Chapter 6. Compared to the SSDP by using simulations, the 2K-SH has slightly higher average total costs and a higher minimum total cost, while both the maximum cost and the standard deviation is the same for both policies.

Table 7.10 shows the average failures and inspections per simulations for the derived policies. The average inspections for both the CBM-O and the PSDP are exactly six as inspections are always carried out in year 1 and with intervals of 9 years. Over the 10 000 simulations, the SSDP suggest more inspections than the 2K-H, and the failure rate is also slightly lower for the SSDP. This implies that the heuristic derives a less conservative policy with a slightly higher risk of failure. Combined with the optimal maintenance decisions, this gives a lower total average cost for the PSDP. We also see that the 2K-SH achieves a lower failure rate than the SSDP and the CBM-O, indicating that the sequential inspection structure helps achieve a lower risk of failure. However, the PSDP makes optimal maintenance decisions given the periodic inspection interval, which seems to be the reason of similar average total cost compared to the heuristic.

Policy	Avg. System Failures	Avg. Inspections
CBM-O	5.58	6.00
PSDP	5.73	6.00
SSDP	5.46	7.58
2K-H	**5.48**	**7.09**

Table 7.10: The table shows system failure rates and average inspections from 10 000 simulations, with the heuristic policy's performance in bold.

7.6 Applicability of results

Throughout this chapter, we see that the Periodic SDP and Sequential SDP suggest solutions that, on average, will perform better than conventional inspection and maintenance policies. However, it is also clear that there are limitations to the applications of our models. This section highlights some of the most interesting properties of the models and their results and indicates which real-life purposes they can, and cannot, prove useful.

7.6.1 Direct use of the solutions

When finding optimal policies, our models compute optimal decisions for all possible future outcomes. Consequently, one does not need to calculate the optimal choice each time one needs to make a decision. This ability strengthens the models' usability for operative purposes, as a technician can check the already avaliable solution when deciding whether or not to maintain a component in the field.

In addition to possible operative applications, the models can be useful on both a tactical and strategic level. Solving for many different power grid lines allows the operator to see them in relation, indicating when different lines should be maintained and which costs can be expected. On a strategic level, the solutions indicate which lines have the most economic significance and should be prioritized.

As Section 7.2.2 points out, end of horizon costs has a considerable influence on the solutions. Leveraging this property, the operator can find optimal policies for special situations, for instance, when planning to take a line out of commission at a given time in the future.

7.6.2 Simplified solutions

While our models solve to optimality, they are unfit for systems with a large number of components. It is clear that simplifications must be made for practical applications of the solutions.

In this thesis, we indicate ways to help bridge the gap between theory and real-life usage. The easiest, naive approach, is to derive condition-based maintenance policies based on the suggested solutions. The CBM-O presented in 7.3.1 is such a policy, and we see that this policy performs better than other CBM-policies set based on convention and "gut feel". Using this approach, we focus on the components with the most significant implications for the ideal policies. We recommend considering components with considerable economic implications first. Then, one can decide on policies for less important components, constrained by the initial solution.

Using more sophisticated approaches is also possible, and in Section 7.5 we present a heuristic that combines the solutions from several smaller systems to approximate the solution provided by the SSDP. The heuristic is significantly faster than the SSDP and provides a solution that outperforms all the conventional policies. While it is not optimal, it performs almost as well as the SDPs. It can be seen as a proof of concept for how to achieve policies for bigger systems that account for dependencies while having acceptable run times.

7.6.3 Understanding the inspection- maintenance relationship

This thesis highlights the interconnection between inspection, maintenance and risk. Section 7.2.3 shows that ideal inspection intervals are subject to the systems state and expected development and the applied maintenance policy. In Section 7.4, we see that the connection between inspection and maintenance is complex. Sometimes, fewer inspections can lead to less maintenance, as the ability to time maintenance correctly is reduced. Other times, a more conservative maintenance policy can be applied, increasing the number of maintenances while inspecting less frequently.

It is tempting to jump to easy, clear cut conclusions. However, for complex problems like the one we study in this thesis, it is valuable to dig deeper to understand the underlying mechanics of how things work. While the problem studied in this thesis cannot be directly applied to a real-life power grid line, studying the results from optimal policies can provide an understanding that improves the ability to derive effective inspection and maintenance policies. To quote the famous saying of Richard Hamming:

"*The purpose of (scientific) computing is insight, not numbers.*" (Hamming, 1962)

Chapter 8

Future Research

Throughout this thesis, we address several interesting topics not included in the described Utility Mast Inspection and Maintenance Problem, and consequently not addressed by the suggested models. This chapter briefly presents some of these topics, suggesting them as starting points for future research.

Considering several dependency types between components

In this thesis, we consider a multi-unit system where components share the setup cost of maintenance when maintained simultaneously. Components may also have structural dependencies. That is, a mast can function properly until a combination of components fails. Furthermore, components can also have probabilistic dependencies, i.e. deterioration of one component may affect another component's transition probabilities. For example, we have a probabilistic dependency between a masts' top hat and its pole. Without the top hat, the mast functions properly, but the pole will rot faster. Including such dependencies in the models is an interesting topic for further research. A suggestion is to assign several deterioration matrices to one component where the matrix used in a calculation depends on the state of another component.

Including imperfect information from inspections

When inspecting a mast, our models receive perfect information about the state of all components. In reality, equipment and human error may result in uncertain information from inspections. Implementing imperfect information is an interesting extension of the problem we describe in this thesis. An option is to consider state probability distributions, rather than actual states returned from inspections. From the literature review in Chapter 3, we know that modelling a partially observable Markov decision process and using point-based solvers is also a promising possibility.

Incorporating different inspection types

In this thesis, we consider one kind of inspection, returning information about all the modelled components. However, various inspection types can be conducted, e.g. with different equipment. Consequently, inspection types can return different utility mast information. For instance, many power grid operators also conduct yearly superficial, aerial inspections, as we discuss in Chapter 2. A development of our model can include several possible inspection decisions, each returning information about subsets of the mast's components.

Further exploring heuristic approaches

To counteract the complexity of large problems, we propose a heuristic incorporating one of our models. The heuristic provides a starting point for future research. As we consider a mast representative for a grid line, reducing the problem complexity has positive implications for the entire power grid. We encourage further research on deriving heuristics based on this thesis's work, and especially extending the use of the one proposed here.

Chapter 9

Concluding Remarks

This Master's thesis proposes a stochastic dynamic programming approach to provide decision support for inspection and maintenance on a power grid line. We formulate two stochastic dynamic programs (SDPs) for inspecting and maintaining a multi-unit utility mast.

One model, the Periodic SDP, considers static inspection intervals, while the other, the Sequential SDP, sets inspection intervals dynamically following an inspection. Both models find optimal policies given the inspection structure. The Sequential SDP outperforms the Periodic SDP because of the ability to make new inspection decisions based on revealed information. We study their solutions using a case with a simplified mast system comprised of four components. The components are critical to the mast's ability to deliver electricity. We use five discrete states to represent the condition of each component. The case has a maximum allowed inspection interval of ten years and considers a planning horizon of 50 years. We benchmark the suggested policies from our two models using simulation and compare them to conventional policies representing typical policies used by power grid operators today. On average, the Periodic SDP achieves 2.2% - 11.9% lower total cost and the Sequential SDP achieves 2.6% - 12.2% lower total cost when simulated over 50 years. From our models' solutions, we also derive an easily applicable condition-based maintenance-policy. This policy performs better than other conventional policies, with 1.0% - 10.7% lower average total cost.

The number of modelled components has a significant impact on the run time of the models. The periodic model use approximately three hours to solve the case with four components, while the sequential use approximately nine and a half hours. We suggest a heuristic approach to enable modelling of larger systems. The heuristic uses the Sequential SDP and solves all possible component-pairs of the mast's components, before joining their optimal policies to derive a policy for the entire multi-unit system. The heuristic solves the four-component case in approximately 75 seconds. Simulations with the heuristic's solution yield promising results. Its average total cost is almost identical to the Periodic SDP's average cost and 0.4% higher than the Sequential SDP's average cost.

This thesis shows that expected power grid operation costs can be reduced by applying sophisticated policies. Such policies should consider dependencies between components and how inspections can be exploited for better maintenance decisions. We suggest further research of some topics, and our models provide a first step for inspection and maintenance optimization of the power grid line. Furthermore, de Jonge and Scarf (2020) suggest that future research considers optimal dynamic scheduling of inspections. This Master's thesis does so by providing a model, the Sequential SDP, that derives an optimal dynamic inspection schedule.

Bibliography

[1] E. Abbasi, M. Fotuhi-Firuzabad and A. Abiri-Jahromi. *Risk based maintenance optimization of overhead distribution networks utilizing priority based dynamic programming*. 2009, pp. 1–11.

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

[3] C. P. Andriotis and K. G. Papakonstantinou. *Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints*. 2020. arXiv: `2007.01380`.

[4] M. J. Armstrong. 'Age repair policies for the machine repair problem'. In: *European Journal of Operational Research* 138.1 (2002), pp. 127–141.

[5] D. Assaf and J. G. Shanthikumar. 'Optimal group maintenance policies with continuous and periodic inspections'. In: *Management Science* 33.11 (1987), pp. 1440–1452.

[6] G. Augland and N.T. Staurvik. *Risikobasert vedlikeholdsplanlegging og fornyelsesstrategi for høyspente luftlinjer i distribusjonsnettet*. 2014.

[7] V. Babishin and S. Taghipour. *Joint maintenance and inspection optimization of a k-out-of-n system*. 2016, pp. 1–6.

[8] S. A. Bakken. 'Mye å spare på linjeinspeksjon'. In: *Energiteknikk* 7 (2019), p. 24.

[9] R. E. Barlow and F. Proschan. *Mathematical theory of reliability*. New York: John Wiley, 1965.

[10] R. Bellman. 'The theory of dynamic programming'. In: *Bulletin of the American Mathematical Society* 60.6 (1954), pp. 503–515.

[11] M. Ben-Daya, U. Kumar and D. N. P. Murthy. *Introduction to maintenance engineering: modelling, optimization and management*. John Wiley & Sons, 2016.

[12] C. Berenguer, C. Chu and A. Grall. 'Inspection and maintenance planning: An application of semi-Markov decision processes'. In: *Journal of intelligent manufacturing* 8.5 (1997), pp. 467–476.

[13] M. D. Berrade, P. A. Scarf and C. A. V. Cavalcante. 'Some insights into the effect of maintenance quality for a protection system'. In: *IEEE Transactions on Reliability* 64.2 (2015), pp. 661–672.

[14] R. S. Bird. 'Tabulation techniques for recursive programs'. In: *ACM Computing Surveys (CSUR)* 12.4 (1980), pp. 403–417.

[15] T. H. A. Blaauw and J. Erikstad. *A stochastic dynamic programming approach to power grid maintenance optimization*. 2019.

[16] S. Bloch-Mercier. 'A preventive maintenance policy with sequential checking procedure for a Markov deteriorating system'. In: *European Journal of Operational Research* 142.3 (2002), pp. 548–576.

[17] J. A. Bloom, C. Feinstein and P. Morris. *Optimal replacement of underground distribution cables*. 2006, pp. 389–393.

[18] E. Byon and Y. Ding. 'Season-dependent condition-based maintenance for a wind turbine using a partially observed Markov decision process'. In: *IEEE Transactions on Power Systems* 25.4 (2010), pp. 1823–1834.

[19] J. H. Cha, M. Finkelstein and G. Levitin. 'On preventive maintenance of systems with lifetimes dependent on a random shock process'. In: *Reliability Engineering & System Safety* 168 (2017), pp. 90–97.

[20] J. Chiang and J. Yuan. 'Optimal maintenance policy for a Markovian system under periodic inspection'. In: *Reliability Engineering & System Safety* 71.2 (2001), pp. 165–172.

[21] C. Chu, J. Proth and P. Wolff. 'Predictive maintenance: The one-unit replacement model'. In: *International Journal of Production Economics* 54.3 (1998), pp. 285–295.

[22] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein. *Introduction to Algorithms, 3rd-edition*. McGraw-Hill, 2009.

[23] W. M. G. Courage, M. P. Nicoreac and P. L. Kempker. *Inspection and maintenance optimization for owt's using bayesian networks*. CRC Press/Balkema, 2017.

[24] C. D. Dao and M. J Zuo. 'Selective maintenance for multistate series systems with s-dependent components'. In: *IEEE Transactions on Reliability* 65.2 (2015), pp. 525–539.

[25] B. de Jonge and P. A. Scarf. 'A review on maintenance optimization'. In: *European Journal of Operational Research* 285.3 (2020), pp. 805–824.

[26] R. Dekker. 'Applications of maintenance optimization models: a review and analysis'. In: *Reliability engineering & system safety* 51.3 (1996), pp. 229–240.

[27] R. Dekker, R. E. Wildeman and F. A. van der Duyn Schouten. 'A review of multi-component maintenance models with economic dependence'. In: *Mathematical methods of operations research* 45.3 (1997), pp. 411–435.

[28] C. Derman. 'On optimal replacement rules when changes of state are Markovian'. In: *Mathematical optimization techniques* 396 (1963), pp. 201–210.

[29] D. Dinh, P. Do and B. Iung. 'Degradation modeling and reliability assessment for a multi-component system with structural dependence'. In: *Computers & Industrial Engineering* (2020), p. 106443.

[30] S. Dreyfus. 'Richard Bellman on the birth of dynamic programming'. In: *Operations Research* 50.1 (2002), pp. 48–51.

[31] P. L. Durango-Cohen and S. M. Madanat. 'Optimization of inspection and maintenance decisions for infrastructure facilities under performance model uncertainty: a quasi-Bayes approach'. In: *Transportation Research Part A: Policy and Practice* 42.8 (2008), pp. 1074–1085.

[32] S. R. Eddy. 'What is a hidden Markov model?' In: *Nature biotechnology* 22.10 (2004), pp. 1315–1316.

[33] B. A. Ellis. 'Condition based maintenance'. In: *The Jethro Project* 10 (2008), pp. 1–5.

[34] H. Ellis, M. Jiang and R. B. Corotis. 'Inspection, maintenance, and repair with partial observability'. In: *Journal of Infrastructure Systems* 1.2 (1995), pp. 92–99.

[35] Energi Norge. *Nettselskapene blir mer effektive*. 2017. URL: `https://www.energinorge.no/fagomrader/stromnett/nyheter/2017/nettselskapene-blir-mer-effektive/` (visited on 24/11/2020).

[36] H. Gao, L. Cui and Q. Qiu. 'Reliability modeling for degradation-shock dependence systems with multiple species of shocks'. In: *Reliability Engineering & System Safety* 185 (2019), pp. 133–143.

[37] H. R. Golmakani and H. Moakedi. 'Periodic inspection optimization model for a multi-component repairable system with failure interaction'. In: *The International Journal of Advanced Manufacturing Technology* 61.1-4 (2012), pp. 295–302.

[38] R. W. Hamming. *Numerical methods for Scientists and Engineers,* McGraw-Hill, 1962.

[39] F. S. Hillier and G. J. Lieberman. *Introduction to operations research*. Tata McGraw-Hill Education, 2012.

[40] A. D. Janjic and D. S. Popovic. 'Selective maintenance schedule of distribution networks based on risk management approach'. In: *IEEE transactions on power systems* 22.2 (2007), pp. 597–604.

[41] E. P. C. Kao. 'Optimal replacement rules when changes of state are semi-Markovian'. In: *Operations Research* 21.6 (1973), pp. 1231–1249.

[42] H. Kawai. 'An optimal ordering and replacement policy of a Markovian deterioration system under incomplete observation: part II'. In: *Journal of the Operations Research Society of Japan* 26.4 (1983), pp. 293–308.

[43] M. C. A. O. Keizer, S. D. P. Flapper and R. H. Teunter. 'Condition-based maintenance policies for systems with multiple dependent components: A review'. In: *European Journal of Operational Research* 261.2 (2017), pp. 405–420.

[44] E. S. Kiel and G. H. Kjølle. *Transmission line unavailability due to correlated threat exposure*. 2019, pp. 1–6.

[45] A. J. King and S. W. Wallace. *Modeling with stochastic programming*. Springer Science & Business Media, 2012.

[46] G. H. Kjølle, I. B. Utne and O. Gjerde. 'Risk analysis of critical infrastructures emphasizing electricity supply and interdependencies'. In: *Reliability Engineering & System Safety* 105 (2012), pp. 80–89.

[47] G.H. Kjølle, K. Samdal, B. Singh and O. A. Kvitastein. 'Customer costs related to interruptions and voltage problems: Methodology and results'. In: *IEEE Transactions on power systems* 23.3 (2008), pp. 1030–1038.

[48] P. Kolesar. 'Minimum cost replacement under Markovian deterioration'. In: *Management Science* 12.9 (1966), pp. 694–706.

[49] J. Korpijärvi and J. Kortelainen. *A dynamic programming model for maintenance of electric distribution system*. Vol. 41. 2009, pp. 636–639.

[50] W. Kuo, V. R. Prasad, F. A. Tillman and C. Hwang. *Optimal reliability design: fundamentals and applications*. Cambridge university press, 2001.

[51] M. Kurt and J. P. Kharoufeh. 'Monotone optimal replacement policies for a Markovian deteriorating system in a controllable environment'. In: *Operations Research Letters* 38.4 (2010), pp. 273–279.

[52] K. Y. Kutucuoglu, J. Hamali, Z. Irani and J. M. Sharp. 'A framework for managing maintenance using performance measurement systems'. In: *International Journal of Operations & Production Management* 21.1/2 (2001), pp. 173–195.

[53] R. Laggoune, A. Chateauneuf and D. Aissani. 'Impact of few failure data on the opportunistic replacement policy for multi-component systems'. In: *Reliability Engineering & System Safety* 95.2 (2010), pp. 108–119.

[54] C. T. Lam and R. H. Yeh. 'Optimal maintenance-policies for deteriorating systems under various maintenance strategies'. In: *IEEE Transactions on reliability* 43.3 (1994), pp. 423–430.

[55] T. Langset, F. Trengereid, K. Samdal and J. Heggset. *Quality dependent revenue caps-a model for quality of supply regulation*. Vol. 6. 1; VOL 6. 2001, pp. 6–4.

[56] G. Leister. *Passenger Car Tires and Wheels: Development-Manufacturing-Application*. Springer, 2018.

[57] E. Levin, R. Pieraccini and W. Eckert. *Using Markov decision process for learning dialogue strategies*. Vol. 1. 1998, pp. 201–204.

[58] H. Li, E. Deloux and L. Dieulle. 'A condition-based maintenance policy for multi-component systems with Lévy copulas dependence'. In: *Reliability Engineering & System Safety* 149 (2016), pp. 44–55.

[59] H. Liao, E. A. Elsayed and L. Chan. 'Maintenance of continuously monitored degrading systems'. In: *European Journal of Operational Research* 175.2 (2006), pp. 821–835.

[60] C. Lim and S. Han. 'A study on development of power grid fault prediction system based on big data and preceding activities to calculate optimal investment cost'. In: *The Korean Data & Information Science Society* 29.3 (2018), pp. 779–794.

[61] B. Liu, R. Yeh, M. Xie and W. Kuo. 'Maintenance scheduling for multicomponent systems with hidden failures'. In: *IEEE Transactions on Reliability* 66.4 (2017), pp. 1280–1292.

[62] X. Liu, J. Li, K.N. Al-Khalifa, A. S. Hamouda, D. W Coit and E. A. Elsayed. 'Condition-based maintenance for continuously monitored degrading systems with multiple failure modes'. In: *IIE transactions* 45.4 (2013), pp. 422–435.

[63] Y. Liu, Y. Chen and T. Jiang. 'Dynamic selective maintenance optimization for multi-state systems over a finite horizon: A deep reinforcement learning approach'. In: *European Journal of Operational Research* 283.1 (2020), pp. 166–181.

[64] D. Lugtigheid, D. Banjevic and A. K. S. Jardine. 'System repairs: When to perform and what to do?' In: *Reliability Engineering & System Safety* 93.4 (2008), pp. 604–615.

[65] J. Lundgren, M. Rönnqvist and P. Värbrand. *Optimization*. Studentlitteratur, 2012.

[66] L. M Maillart. 'Maintenance policies for systems with condition monitoring and obvious failures'. In: *Iie Transactions* 38.6 (2006), pp. 463–475.

[67] J. J. McCall. 'Maintenance policies for stochastically failing equipment: a survey'. In: *Management science* 11.5 (1965), pp. 493–524.

[68] M. R. K. Mes and A. P. Rivera. 'Approximate dynamic programming by practical examples'. In: *Markov Decision Processes in Practice*. Springer, 2017, pp. 63–101.

[69] Ministry of Petroleum and Energy. *The Electricity Grid*. 2019. URL: `https://energifaktanorge.no/en/norsk-energiforsyning/kraftnett/` (visited on 15/09/2020).

[70] H. Mjølnerød. *Introducing WAI - Power grid maintenance entering the digital age*. 2019. URL: `https://www.wiseline.no/index.php/nyheter/57-introducing-wai` (visited on 26/01/2021).

[71] P. G. Morato, K. G. Papakonstantinou, C. P. Andriotis, J. S. Nielsen and P. Rigo. *Optimal inspection and maintenance planning for deteriorating structures through dynamic Bayesian networks and Markov decision processes*. 2020. arXiv: `2009.04547`.

[72] P. Muchiri, L. Pintelon, L. Gelders and H. Martin. 'Development of maintenance function performance measurement framework and indicators'. In: *International Journal of Production Economics* 131.1 (2011), pp. 295–302.

[73] D. N. P. Murthy and D. G. Nguyen. 'Study of two-component system with failure interaction'. In: *Naval Research Logistics Quarterly* 32.2 (1985), pp. 239–247.

[74] T. Nakagawa and S. Mizutani. 'A summary of maintenance policies for a finite interval'. In: *Reliability Engineering & System Safety* 94.1 (2009), pp. 89–96.

[75] S. Nakajima. 'Introduction to TPM: total productive maintenance'. In: *Productivity Press, Inc., 1988,* (1988), p. 129.

[76] R. B. Nelsen. 'Consequences of the memoryless property for random variables'. In: *The American Mathematical Monthly* 94.10 (1987), pp. 981–984.

[77] M. L. Neves, L. P. Santiago and C. A. Maia. 'A condition-based maintenance policy and input parameters estimation for deteriorating systems under periodic inspection'. In: *Computers & Industrial Engineering* 61.3 (2011), pp. 503–511.

[78] R. P. Nicolai. *Maintenance models for systems subject to measurable deterioration*. 420. Rozenberg Publishers, 2008.

[79] R. P. Nicolai and R. Dekker. 'Optimal maintenance of multi-component systems: a review'. In: *Complex system maintenance handbook*. Springer, 2008, pp. 263–286.

[80] Nordlandsnett AS. *Vedlikehold - Hva gjør vi?* 2019. URL: `https://nordlandsnett.no/stromnettet/?Article=135` (visited on 14/09/2020).

[81] M. Nordnes. *TIlstandskontroll av kraftnett håndbok - kraftledning*. Vol. 338. Middelthuns gate 27, Oslo, Norway: EnergiNorge AS, 2011. ISBN: 978-82-432-0666-3.

[82] NVE. *Hvem bestemmer prisen på nettleien?* 2019. URL: `https://www.skageraknett.no/priser/hvem-bestemmer-prisen-pa-nettleien-article808-964.html` (visited on 19/10/2020).

[83] NVE. *Nøkkeltall for nettselskapene*. 2020. URL: `https://www.nve.no/reguleringsmyndigheten/okonomisk-regulering-av-nettselskap/nokkeltall-for-nettselskapene/` (visited on 23/01/2020).

[84] NVE. *Sammendrag av nøkkeltallene for nettselskapene*. 2020. URL: `https://www.nve.no/media/11301/sammendrag.pdf` (visited on 23/01/2020).

[85] T. Onoufriou and D. M. Frangopol. 'Reliability-based inspection optimization of complex structures: a brief retrospective'. In: *Computers & structures* 80.12 (2002), pp. 1133–1144.

[86] K. G. Papakonstantinou and M. Shinozuka. 'Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part I: Theory'. In: *Reliability Engineering & System Safety* 130 (2014), pp. 202–213.

[87] V. A. Petrushin. *Hidden markov models: Fundamentals and applications*. 2000.

[88] W. P. Pierskalla and J. A. Voelker. 'A survey of maintenance models: the control and surveillance of deteriorating systems'. In: *Naval Research Logistics Quarterly* 23.3 (1976), pp. 353–388.

[89] L. Pintelon and L. F. Gelders. 'Maintenance management decision making'. In: *European journal of operational research* 58.3 (1992), pp. 301–317.

[90] L. Pintelon and F. van Puyvelde. *Maintenance decision making*. Acco, 2006.

[91] W. B. Powell. 'Perspectives of approximate dynamic programming'. In: *Annals of Operations Research* 241.1-2 (2016), pp. 319–356.

[92] Rasjonell Elektrisk Nettvirksomhet AS. *Pris standard stolpeskift*. 2019. URL: `https://www.ren.no/retningslinjer-og-verktoy` (visited on 07/11/2020).

[93] E. Reiten, K. Bjella and L. Sørgard. *Et bedre organisert strømnett*. Ministry of Petroleum and Energy, 2014.

[94] S. Riibe and H. Weyergang-Nielsen. *Kraftoverføringens kulturminner*. The Norwegian Water Resources and Energy Directorate, 2010.

[95] S. M. Ross. 'A Markovian replacement model with a generalization to include stocking'. In: *Management Science* 15.11 (1969), pp. 702–715.

[96] K.A. Rosvold. 'Regionalnett'. In: *Store Norske Leksikon* (2020). URL: `https://snl.no/regionalnett` (visited on 15/09/2020).

[97] S. Sachan, C. Zhou, G. Bevan and B. Alkali. *Cost effective replacement of power cables by stochastic dynamic programming approach*. 2016, pp. 299–302.

[98] M. Shafiee and J. D. Sørensen. 'Maintenance optimization and inspection planning of wind energy assets: Models, methods and strategies'. In: *Reliability Engineering & System Safety* 192 (2019), p. 105993.

[99] A. Sharma, G. S. Yadava and S. G. Deshmukh. 'A literature review and future perspectives on maintenance optimization'. In: *Journal of Quality in Maintenance Engineering* 17.1 (2011), pp. 5–25.

[100] J. Shen, A. Elwany and L. Cui. 'Reliability analysis for multi-component systems with degradation interaction and categorized shocks'. In: *Applied Mathematical Modelling* 56 (2018), pp. 487–500.

[101] J. Shen, J. Hu and Y. Ma. 'Two preventive replacement strategies for systems with protective auxiliary parts subject to degradation and economic dependence'. In: *Reliability Engineering & System Safety* 204 (2020), p. 107144.

[102] Y. S. Sherif and M. L. Smith. 'Optimal maintenance models for systems subject to failure–a review'. In: *Naval research logistics quarterly* 28.1 (1981), pp. 47–74.

[103] S. Sheu, T. Liu, Z. Zhang and H. Tsai. 'The generalized age maintenance policies with random working times'. In: *Reliability Engineering & System Safety* 169 (2018), pp. 503–514.

[104] E Solvang and J. Foros. *Analyse av fornyelsesbehov i kraftnett*. Vol. 442. Middelthuns gate 27, Oslo, Norway: EnergiNorge AS, 2019. ISBN: 978-82-436-1061-3.

[105] E. J. Sondik. *The optimal control of partially observable Markov processes.* Tech. rep. Stanford Univ Calif Stanford Electronics Labs, 1971.

[106] Statnett. *Årsstatistikk 2018 - Driftsforstyrrelser, feil og planlagte utkoplinger i 1-22 kV-nettet*. Statnett, 2019.

[107] B. Stene, T. M. Sneve and K. Brekke. *Aldersfordeling for komponenter i kraftsystemet Levetid og behov for reinvesteringer*. Vol. 100. Middelthuns gate 29, Oslo, Norway: The Norwegian Water Resources and Energy Directorate, 2005. ISBN: 82-410-0542-3.

[108] The Local Electrical Supervisory. *Helse, miljø og sikkerhet: Internkontroll for elektriske anlegg og utstyr*. 2020. URL: `https://www.elsikkerhetsportalen.no/wp-content/uploads/2018/07/Internkontroll-for-elanlegg-med-komp.pdf` (visited on 18/10/2020).

[109] L. Syvertsen, M. Lagergren, J. Kristiansen and P. Melvær. *Analyse av utviklingen i nøkkeltall for strømnettvirksomheten*. The Norwegian Water Resources and Energy Directorate, 2018.

[110] S. Taghipour and D. Banjevic. 'Periodic inspection optimization models for a repairable system subject to hidden failures'. In: *IEEE Transactions on Reliability* 60.1 (2011), pp. 275–285.

[111] L. C. Thomas. 'A survey of maintenance and replacement models for maintainability and reliability of multi-item systems'. In: *Reliability Engineering* 16.4 (1986), pp. 297–309.

[112] L. C. Thomas, D. P. Gaver and P. A. Jacobs. 'Inspection models and their application'. In: *IMA Journal of Management Mathematics* 3.4 (1991), pp. 283–303.

[113] U.S. Department of Transportation's Federal Highway Administration. *Average Annual Miles per Driver by Age Group*. 2018. URL: `https://www.fhwa.dot.gov/ohim/onh00/bar8.htm` (visited on 10/01/2021).

[114] United States Departement of Defence. *Department of defense dictionary of military and associated terms*. 2019.

[115] A. van Horenbeek, L. Pintelon and P. Muchiri. 'Maintenance optimization models and criteria'. In: *International Journal of System Assurance Engineering and Management* 1.3 (2010), pp. 189–200.

[116] C. van Oosterom, H. Peng and G. van Houtum. 'Maintenance optimization for a Markovian deteriorating system with population heterogeneity'. In: *IISE Transactions* 49.1 (2017), pp. 96–109.

[117] A. K. Verma, A. Srividya and R. S. P. Gaonkar. 'Deciding dynamic inspection frequency: A fuzzy set approach'. In: *International Journal of Performability Engineering* 4.1 (2006), pp. 45–59.

[118] H. Wang. 'A survey of maintenance policies of deteriorating systems'. In: *European journal of operational research* 139 (2002), pp. 469–489.

[119] W. Wang. 'A two-stage prognosis model in condition based maintenance'. In: *European journal of operational research* 182.3 (2007), pp. 1177–1187.

[120] Y. Wang and H. Pham. 'A multi-objective optimization of imperfect preventive maintenance policy for dependent competing risk systems with hidden failure'. In: *IEEE Transactions on Reliability* 60.4 (2011), pp. 770–781.

[121] Wiseline AS. *Produkter/Tjenester*. 2017. URL: `https://www.wiseline.no/index.php/produkter/` (visited on 26/01/2021).

[122] F. Wu, S. A. Niknam and J. E. Kobza. 'A cost effective degradation-based maintenance strategy under imperfect repair'. In: *Reliability Engineering & System Safety* 144 (2015), pp. 234–243.

[123] S. Wu, Y. Chen, Q. Wu and Z. Wang. 'Linking component importance to optimisation of preventive maintenance policy'. In: *Reliability Engineering & System Safety* 146 (2016), pp. 26–32.

[124] H. Xiao and M. Cao. 'Balancing the demand and supply of a power grid system via reliability modeling and maintenance optimization'. In: *Energy* 210 (2020), p. 118470.

[125] L. Yang, Y. Zhao and X. Ma. 'Multi-level maintenance strategy of deteriorating systems subject to two-stage inspection'. In: *Computers & Industrial Engineering* 118 (2018), pp. 160–169.

[126] Y. Zhou, T. R. Lin, Y. Sun and L. Ma. 'Maintenance optimisation of a parallel-series system with stochastic and economic dependence under limited maintenance capacity'. In: *Reliability Engineering & System Safety* 155 (2016), pp. 137–146.

Blaauw, Erikstad

Power Grid Inspection and Maintenance Optimization

NTNU
Norwegian University of
Science and Technology