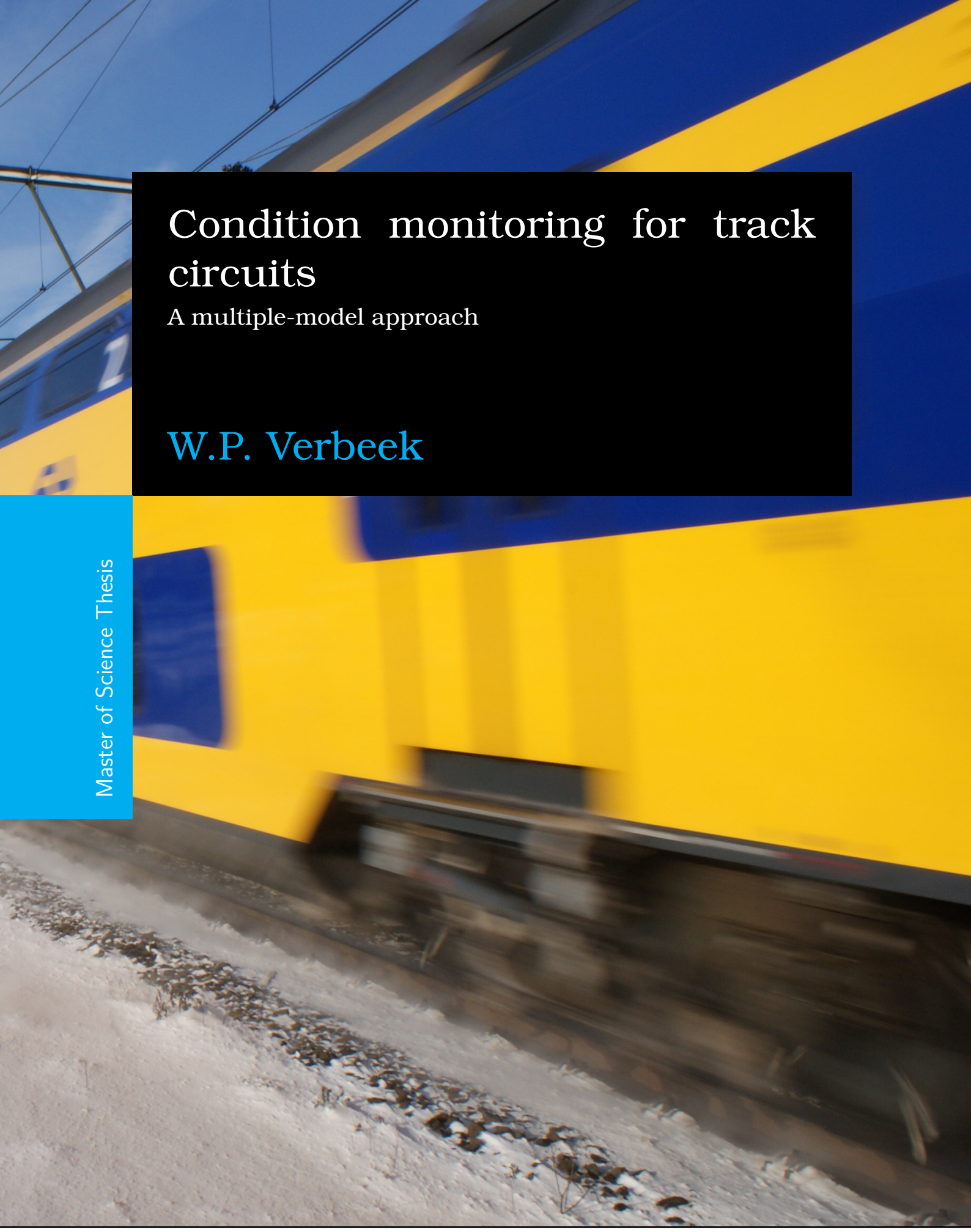


Condition monitoring for track circuits

A multiple-model approach

W.P. Verbeek

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Condition monitoring for track circuits

A multiple-model approach

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

W.P. Verbeek

September 28, 2015

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Source of the cover image: [30].

Delft University of Technology
Department of
Delft Center for Systems and Control (dcsc)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

Condition monitoring for track circuits

by

W.P. Verbeek

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: September 28, 2015

Supervisor(s):

prof.dr.ir. B. De Schutter

ir. K.A.J. Verbert

Reader(s):

Dr.-Ing. S. Wahls

dr. A.A. Núñez Vicencio

Abstract

Preventive maintenance is a maintenance strategy where maintenance is performed prior to failure. A promising preventive maintenance strategy is condition-based maintenance, which schedules maintenance based on the current or predicted future condition of the assets. An adequate implementation of this strategy avoids breakdowns and reduces maintenance costs. Condition-based maintenance requires condition monitoring, which is the process of translating sensor data into information on the health state of the monitored objects. Condition monitoring consists of two tasks: classifying what faults are present and predicting the remaining useful life of the monitored asset. Over the past few years, ProRail (the Dutch rail infrastructure manager) has increased the amount of monitoring devices, but does not yet use the collected data optimally for condition monitoring. In this thesis, we aim at closing this gap and present a condition monitoring approach that effectively combines the available sensor data with system knowledge.

In particular, we focus on condition monitoring for track circuits, which are devices used to detect whether a train is present on a section of the track. To classify faults and to predict the remaining useful life of track circuits, a multiple-model approach is used. A multiple-model approach can be seen as a state estimation and prediction method for a specific class of hybrid systems, where for each mode, the system can be described using a continuous linear state space model. This modeling methodology is well suited to describe systems subject to structural changes, such as faults. We show that the track circuit case fits into the multiple-model framework and we develop models for each fault state of the track circuit based on system knowledge and monitoring data. To predict the remaining useful life, a Monte Carlo approach is used in order to take known uncertainties into account and to provide decision makers with estimates of the prediction uncertainty.

In a simulation-based case study, the approach is able to correctly classify 99 % of the cases and therewith outperforms a long short-term neural network developed for the same problem, which is a state of the art condition monitoring method for track circuits. Furthermore, an average prediction accuracy for the remaining useful life of 81 % is achieved. In addition, the approach is able to quantify the uncertainty around these predictions and the underlying factors causing the uncertainty.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisors Bart De Schutter and Kim Verbert for their assistance during my thesis project. Their critical reflections on the methods and directions I proposed were very valuable. Often these reflections were brought with a lot of humor. For the comparison between a multiple-model approach and a neural network, I have collaborated a lot with Tim de Bruin and hereby I would like to acknowledge his contribution. The discussions about the approaches and the criteria for performance evaluation I had with Tim were always fun. Lastly, I would like to thank Robert Vrees for reading a draft version of this thesis and giving good advises on further improvement.

Delft, University of Technology W.P. Verbeek
September 28, 2015

"I can live with doubt, and uncertainty, and not knowing. I think it's much more interesting to live not knowing than to have answers which might be wrong."

— *Richard Feynman* [8]

Chapter 1

Introduction

On the 19th of February 2014, ProRail, the Dutch rail infrastructure manager, had to perform emergency repairs on four switches [43]. In the evening, no train traffic was possible between Rotterdam and the Hague, leading to large delays for a lot of passengers. These delays might have been avoided, if ProRail would have used a better maintenance strategy. Condition-based maintenance is such a maintenance strategy. The concept of condition-based maintenance is to perform preventive maintenance based on the actual or predicted future condition of physical assets. In this particular case, condition-based maintenance would have probably lead to scheduled repairs for the four points at different times. This would have avoided the disruption and the corresponding delays for the passengers.

In order to perform condition-based maintenance, condition monitoring is required. Condition monitoring is the process of translating sensor data into information on the health state of the monitored objects. Over the past few years, ProRail has increased the amount of monitoring devices, but does not yet use the collected data optimally for condition monitoring. In this thesis, we aim at closing this gap and present a condition monitoring approach for track circuits that effectively combines the available sensor data with system knowledge.

1-1 Problem description

In this thesis, condition monitoring of track circuits is studied. A track circuit is a device to detect whether a train is present on a section. The track circuit working principle is based on a current flowing through the rails. When a train is present on a section, the axle of the train leads to a short circuit. This is detected using a relay, which triggers a signal to display that the section is occupied. Track circuits can fail due to various causes, such as mechanical defects, rail contamination, or ballast degradation. Before a track circuit fails, the track circuit is often already deteriorating, but still functioning. In that case, we say that the track circuit suffers from a fault. The condition monitoring problem of track circuits consists of two subproblems. The first subproblem is to detect whether a fault is present at a track circuit and to classify this fault. This is called diagnostics. The second subproblem

is to predict at what time the detected fault will lead to a track circuit failure. This is called prognostics.

1-2 Goal

In this project, we aim at developing a condition monitoring solution that addresses both the diagnostic and the prognostic subproblem. Next to developing a theoretical solution, we also aim at implementing the solution and validating the proposed approach using a simulation-based case study. These considerations lead to the following project goal:

> *"The aim of the research is to develop and validate methods to classify faults, predict the system degradation behavior, and predict the corresponding moment of functional system failure of track circuits."*

1-3 Approach

To solve the track circuit condition monitoring problem, we propose a multiple-model approach. A multiple-model approach can be seen as a state estimation and prediction method for a specific class of hybrid systems, where for each mode, the system can be described using a (possibly different) continuous linear state space model. We choose this approach, since faults can be seen as structural changes in the system, which are well-described using this class of systems. In this thesis, we study how the multiple-model approach can be used for condition monitoring and show how the track circuit case fits into this framework. In addition, we perform a case study and implement the proposed condition monitoring solution for a small railway network with track circuits. We evaluate the performance of the condition monitoring program using a simulation-based case study.

1-4 Relevance

The studied problem is highly relevant from both a practical and a scientific point of view. The practical relevance stems, first of all, from the fact that the Human Environment and Transport Inspectorate "is concerned about the inadequate availability of object information and the poor view on the true state of maintenance, since this might lead to hidden defects" [26][1]. The goal of this thesis project is to provide accurate information about the true state of maintenance for one specific type of objects: track circuits. Furthermore, in 2014 the customer satisfaction level about the on-time performance of ProRail and Dutch Railways (NS) was still below the targets set by the Dutch government [39]. Reducing the amount of emergency repairs, will lead to an increased customer satisfaction level. Lastly, it was shown in [48] that the availability of track circuits can be improved, compared to the current practice

[1] Translated from Dutch. Original text: "Ook is de inspectie bezorgd over de gebrekkige beschikbaarheid van objectgegevens en het slechte zicht op de werkelijke staat van het onderhoud, omdat dit kans geeft op verborgen gebreken."

in Sweden, if a condition monitoring program is employed that can detect more than 60 % of the faults.

From a scientific point of view, the study is relevant since condition monitoring for track circuits is not widely studied. The existing studies focus on fault classification and do not address prognostics [12, 13, 20, 45, 63]. Furthermore, often these diagnostic studies make use of a sensor measuring the current flowing through the train axle [13, 20, 45], while we study condition monitoring using the current flowing through the relay of the track circuit and rain sensor data. Our approach is able to perform condition monitoring continuously, where these methods require that a train with specialized measurement equipment runs over the monitored section in order to update information on the current health state. Furthermore, since measurements of the current flowing through the relay of the track circuit are already available for a lot of track circuits and rain sensors can be are relatively inexpensive, our approach leads to lower installation and monitoring costs.

1-5 Organization of this thesis

The rest of this thesis is organized as follows. We start in Chapter 2 with explaining the track circuit working principle and introducing the concepts of condition-based maintenance and condition monitoring. Furthermore, we give an overview of methods that can be used for condition monitoring and motivate the choice for a multiple-model approach. In Chapter 3, we discuss multiple-model state estimation and prediction and make choices with respect to the techniques used. In Chapter 4, we show how the multiple-model approach can be applied to track circuits. In Chapter 5, we test the proposed approach in a simulation-based case study of a small railway network with track circuits and present and discuss the achieved results. Furthermore, we compare the diagnostic performance of our solution with that of a neural network developed for the same purpose [19]. In Chapter 6, we summarize our main findings, discuss the limitations of our research and give recommendations for future work.

Chapter 2

Preliminaries

In this chapter, we give some background on track circuits (Section 2-1) and introduce the concepts condition-based maintenance (Section 2-2) and condition monitoring (Section 2-3). Furthermore, we give an overview of the most widely-used methods for condition monitoring and discuss why a multiple-model method is well-suited to our problem in Section 2-4. Finally, we give an overview of existing studies on condition monitoring for track circuits in Section 2-5. The chapter is summarized in Section 2-6.

2-1 Track circuits

A track circuit is a simple device used to detect whether a train is present on a section. In this section, we outline the track circuit working principle in Section 2-1-1 and faults that can occur on track circuits in Section 2-1-2.

2-1-1 Working principle

The most simple track circuit is the direct current track circuit (Figure 2-1). The direct current track circuit works as follows. Power is applied on one side of the section (the transmitter in Figure 2-1). On the other side of the section, a relay is mounted (the receiver in Figure 2-1). When no train is present, the current flows through the relay, which triggers a signal to display that the section is clear (left in Figure 2-1). When a train is present on the section, the axles of the train function as a shunt and short circuits the rails (right in Figure 2-1). Due to the internal resistance in the transmitter, little current flows through the relay, which triggers the signal to display that the section is occupied. If the power supply fails, a disconnection in the wiring is present or a short circuit occurs, this leads to a false train detection, which is fail safe.

In the Netherlands, the rails are used to transport the return current for traction of trains. Therefore alternating current track circuits have to be used, whereby the return current can

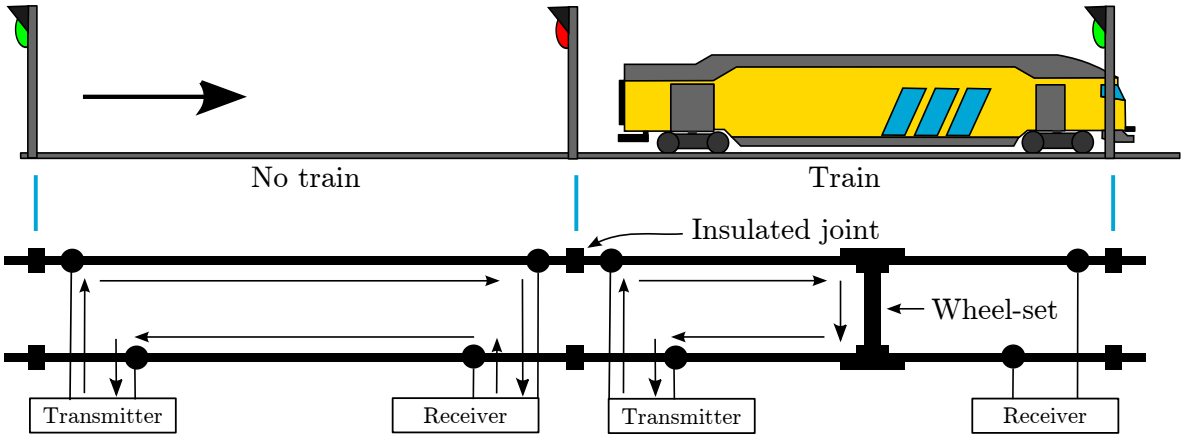


Figure 2-1: The track circuit working principle [19].

pass the joints by an impedance bond. This impedance bond blocks track circuit signals, but permits return currents. In the Netherlands, a 110 V alternating current with a frequency of 75 Hz is applied to the track [68]. A B2 vane relay is used with a local alternating current feed and a track circuit feed as inputs. The relay does only energize when the phase and frequency of the local current and track circuit current match, to ensure the fail safe principle. In modern railway systems, track circuits are connected to railway operating systems [70]. In that case, cables from the track circuit to a nearby railway operating system are used to transport information on the state of the track circuit. Often, a binary signal representing the operating mode of the track circuit (clear or occupied) is used, but sometimes measurements of the current flowing through the relay of the track circuit are transmitted to the railway operating system [55]. The binary signal is useful for railway operators, while the latter can be used for condition monitoring.

In Figure 2-2, the relation between the current flowing through the receiver I_t and the signal displayed to the train driver is shown. When the relay at the receiver is energized, the signal indicates a clear section. The relay needs a minimum current of γ_2 in order to switch from the occupied indication to clear. When a train is present on the track, the signal should report the section as occupied. The relay needs a current lower than γ_1, in order to de-energize the relay. When the relay is de-energized, the signal shows that the section is occupied. Track circuits are designed to have a current on free sections above the threshold α_2 and a current below the threshold α_1 when a train is present on the section. When $\alpha_1 \leq I_t \leq \gamma_1$ and a train is present at the section, the system still correctly reports the section as occupied. In this case a fault is present, but there is no failure. Likewise, if the current $\gamma_2 \leq I_t \leq \alpha_2$ when no train is present on the section, we talk about a fault.

When the current I_t is above γ_1 when a train is present or below γ_2 when the section is free, this is considered a failure of the track circuit. Two types of failures can be distinguished for track circuits: false positive failures and false negative failures. False positive failures take place when no train is present at the section, but due to problems the current flowing through the relay is below γ_2. In that case, the signal indicates the section as occupied, while it is actually free. When a train is present on the section, but due to a problem the current flowing through the relay is above γ_1, a false negative failure is present and the section is accidentally reported as free. False negative failures are the most dangerous type of failures.

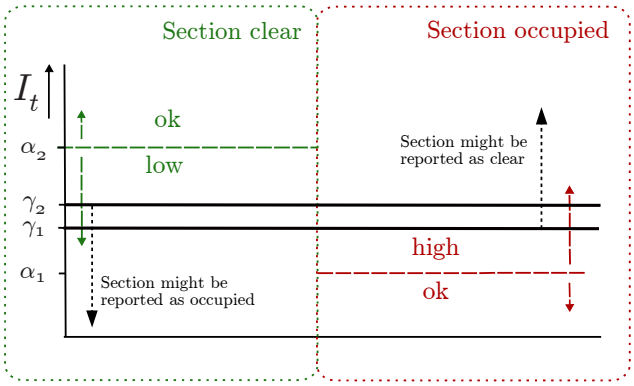


Figure 2-2: Relation between the current I_t flowing through the relay and the system state [79].

Table 2-1: Fault characteristics [79].

Fault (θ)	Problem	Cause	Potential future failure	Current	Spatial influence	Degradation behavior
0	-	Healthy state	-	ok	-	-
1	Train shunt imperfection	Rail contamination	False negative	high	$D_1 \vee D_2 \vee D_4$	-
2		Lightweight train	False negative	high	D_3	-
3	Insulation imperfection	Insulated joint defect	False positive	low	D_1	L $\vee$ E
4		Conductive object	False positive	low	D_1	A
5	Rail conductance impairment	Mechanical defect	False positive	low	D_1	E
6		Electrical disturbance	False positive	low	D_2	I
7	Ballast condition	Ballast degradation	False positive	low	$D_1 \vee D_2$	L $\vee$ E
8		Ballast variation	False positive	low $\vee$ ok $\vee$ high	D_4	A$\vee$ L $\vee$ E $\vee$ I

2-1-2 Faults

Track circuits can suffer from various faults [79]. In this thesis, we constrain us to the faults as listed in Table 2-1. For adequate fault diagnosis/prognosis, we have to understand how the different faults are reflected in the current flowing through the relay at the receiver. To distinguish between the different faults, we particularly consider how the fault

- directly affects its own current signal (column 'Current' in Table 2-1),
- evolves over time (column 'Degradation behavior' in Table 2-1),
- affects nearby track circuits (column 'Spatial influence' in Table 2-1).

Faults may lead to a higher or lower current than prescribed, depending on the type of the fault. In general, faults that could lead to a false negative failure, obey a higher current than prescribed on an occupied section. In contrast, faults that might lead to a false positive failure cause a lower than prescribed clear-section current.

In addition, we follow [19] and adopt the notion of fault intensity[1]. The fault intensity is defined as the percentage of current change. If the fault has no effect on the measured

[1] In [19] this is called fault severity.

current, its intensity is 0 %. When the decrease in current is so large that it causes a failure ($I_t = \gamma_2$) we say the fault intensity is 100 %. Degradation behavior is defined as how the fault intensity evolves from 0 % to 100 % over time. Track circuit faults obey four different types of degradation behavior [79]:

- A: Abrupt degradation behavior; the fault suddenly occurs and the fault intensity does not increase any further.
- L: Linear degradation behavior; the fault intensity increases approximately linearly over time.
- E: Exponential degradation behavior; the fault intensity increases more than linearly by time.
- I: Intermittent degradation behavior; the fault suddenly appears sometimes and later suddenly disappears.

Some of the faults that can be present at a track circuit may influence track circuits at other sections as well. The following types of spatial influence are defined for track circuit faults [79]:

- D_1: The fault influences one specific section
- D_2: The fault influences track circuits on the same track
- D_3: The fault influences track circuits along the path of a specific train
- D_4: The fault influences all nearby sections

When no fault is present, the system is healthy and we denote this with $\theta = 0$. Two types of faults could lead to a false negative failure. Both faults can be seen as train shunt imperfection problems. Train shunt imperfection leads to a higher current flowing through the relay, since the short circuit is partly obstructed due to the train shunt imperfection. Train shunt imperfection problems might be caused by rail contamination ($\theta = 1$), for example leaves on the rails, or by lightweight trains ($\theta = 2$). Rail contamination might be caused by factors that are present on all nearby sections, on one track, or only on one section. The exact form of degradation behavior is unknown. Lightweight trains do not have enough mass to press the wheels on the rails in order to make a good shunt. Lightweight train faults are present on track sections located on the path of the train.

Insulation imperfection problems might eventually lead to false positive failures. Insulation imperfection can be caused by insulated joint defects ($\theta = 3$) or conductive objects placed over the joints ($\theta = 4$). Insulated joint defects allow the current to flow to adjacent track circuits. Fortunately, track circuits are designed to be fail safe and insulated joint defects do not lead to a false negative on adjacent track circuits, since phases of adjacent sections are reversed and the relays can only be energized using in-phase currents. Insulated joint defect may develop linearly or exponentially over time. Conductive objects placed over the joints are another cause of insulation imperfection. Conductive objects cause an abrupt fault that only affects one specific section.

Rail conductance impairment might be caused by mechanical defects ($\theta = 5$) or electrical disturbances ($\theta = 6$). These faults lead to a lower clear-section current. Mechanical defects may cause rail conductance impairment. The severity of these type of faults increases exponentially over time and only affects one section. Electrical disturbances might be caused by high traction currents, which influence all sections on the same track and are assumed to be present intermittently or abruptly.

The ballast condition might influence the track circuit working as well. Ballast is the foundation between the rails. The condition of the ballast might decrease, which could lead to a lower resistance between the rails and consequently a lower current flowing through the receiving relay. When the condition of the ballast is structurally decreasing, we talk about ballast degradation ($\theta = 7$). When the ballast resistance varies over time due to environmental circumstances, we talk about ballast variation ($\theta = 8$). Ballast degradation might be present on all sections on a track or only on one specific section and the degradation behavior might be linear or exponential. The behavior of ballast variation can take various forms, but it is assumed that environmental influences affect all nearby sections.

2-2 Condition-based maintenance

Condition-based maintenance and time-based maintenance are two maintenance strategies that have gained a lot of attention in the literature [87]. Condition-based maintenance is defined as [31] "a decision making strategy where the decision to perform maintenance is reached by observing the 'condition' of the system and/or its components. The condition of a system is quantified by parameters that are continuously monitored and are system or application specific."

The alternative to condition-based maintenance is time-based maintenance, whereby preventive maintenance is carried out periodically. In that case, decisions are made based on failure time data or used-based data [2]. In time-based maintenance it is assumed that the failure characteristics are known and that the system degrades under normal usage [52].

There are two main advantages of condition-based maintenance over time-based maintenance. First of all, condition-based maintenance reduces the number of unnecessary maintenance operations, since action is only taken when evidence exists of abnormal behavior [52]. This can lead to significant maintenance cost reduction [27]. Second, condition-based maintenance leads to a substantially lower number of unexpected breakdowns, since problems can be detected long before the end of the designed life and the resulting failures can be avoided [46]. Of course, disadvantages exist as well, such as the high investment costs associated with condition-based maintenance and the additional skills that might be required in order to effectively implement such a policy [46].

2-3 Condition monitoring

To perform condition-based maintenance, information is required on the condition of the physical assets. The process of obtaining this information is called condition monitoring. Condition monitoring is a concept dating back to the development of the first machines, where human senses where used to monitor the state of the machinery [18]. Nowadays, sensors are used to measure degradation behavior. Condition monitoring consists of two tasks. One task is to detect whether a fault is present and to classify that fault; we call this diagnostics. The second task is to predict at what time the detected fault will lead to failure; this is called prognostics [9]. Essentially, one could see diagnostics as a state estimation problem and prognostics as a prediction problem.

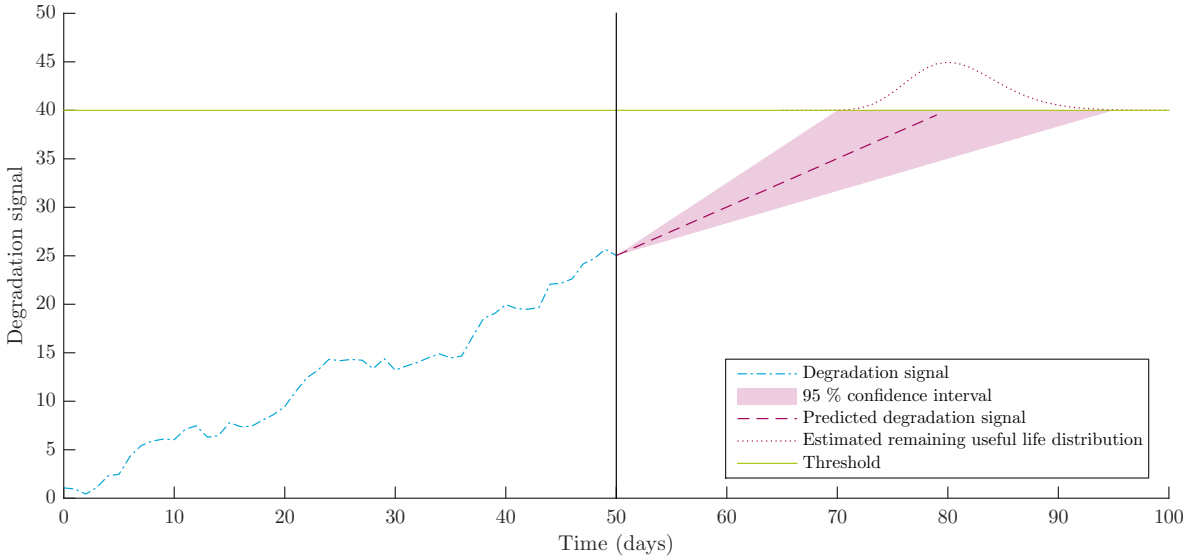


Figure 2-3: Concept of prognostics.

For diagnostics, condition monitoring data is used to determine the state of the system. In its simplest form, a diagnostic method monitors a condition monitoring signal and triggers an alarm when the signal crosses a threshold. The aim of prognostics is to predict at what time the fault that is present will lead to a failure. In general, this is done by predicting the time a degradation signal will cross a failure threshold [86]. When the failure threshold is directly related to a physical failure, this is called a hard threshold. When the threshold is an arbitrary value denoting the time maintenance action should be undertaken, this is called a soft threshold [14]. A degradation signal is an indicator of the health state of the system. In some cases, the degradation signal is observed directly and in other cases only measurements related to the degradation signal are available. In the first case, we talk about direct condition monitoring; the latter case is called indirect condition monitoring [65].

The concept of prognostics is shown in Figure 2-3. With t we denote the time step. In the figure, we see a degradation signal up till $t = 50$ days (the blue dash-dotted line). At $t = 50$ days, the time the signal will cross the failure threshold is predicted based on the degradation signal up till time $t = 50$ days. The time that is left until the degradation signal crosses the threshold, is called the remaining useful life. Predictions for the remaining useful life can be made using many different techniques, ranging from expert systems [6] to gamma process models [75]. In Section 2-4, we discuss the most important methods. For some degradation models no closed-form expressions for the remaining useful life distribution are available [35]. In those cases, one has to resort to an approximation of the remaining useful life distribution (for example using a Monte Carlo approach) or a point prediction of the remaining useful life.

2-4 Diagnostic and prognostic methods

Due to increasing system complexity and a higher demand of system reliability, the interest in condition monitoring has increased during the last two decades [34]. Most research was focused on diagnostics, which has found widespread application, from process industry [76,

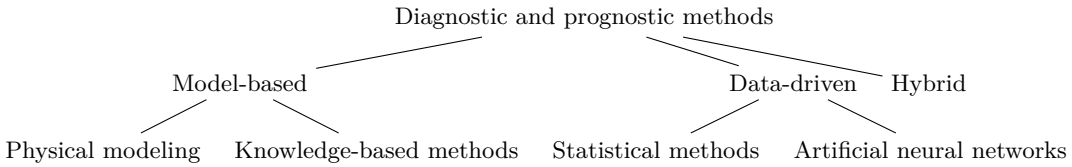


Figure 2-4: Diagnostic and prognostic methods.

77, 78] to rotating machinery [50]. Prognostics is a relatively new area [21]. The literature on prognostics is very application-specific, but in [52] and [66] an overview has been given of the most well-known methodologies.

Diagnostic and prognostic methods are often categorized into two classes (see Figure 2-4): model-based and data-driven approaches. Next to these two main classes, combining model-based techniques with data-driven practices is increasingly popular [52]. These approaches are called hybrid approaches. The model-based approaches can be based on quantitative knowledge of the process (physical modeling approaches) or on qualitative knowledge (knowledge-based methods). Data-driven methods can be divided into statistical methods and approaches using artificial neural networks.

2-4-1 Model-based methods

In physical modeling approaches, it is assumed that mathematical models can be build based on first principles [52]. Residual generation is at "the heart" of those physical modeling approaches [31]. Residuals are the differences between the physical model, based on first principles, and sensor measurements. For diagnostics, the idea is that under normal circumstances these residuals are small and that a fault leads to large residuals. Faults can be detected using statistical techniques (in its simplest form thresholding). The use of physical modeling approaches is discussed in [49]. In this study, special attention is given to robustness. With robustness it is meant that the physical modeling approach should be sensitive to faults, but insensitive to uncertainty. In addition, cases studies are presented in this paper where a diagnostic physical modeling approach is applied to a jet engine, a pumping system and an AC-drive system. Sometimes the physical deterioration process is known and can be used for prognostics. However, often this information is not completely available. In those cases, often physical modeling is combined with statistical techniques. A good example is [10]. In this study, on-line parameter estimation was performed for a model of flight control actuators. As fault features the researchers use the difference between the estimated parameters and the baseline (healthy) model parameters. For prognostics, the researchers relied on a statistical trending method called double exponential smoothing.

Accurate mathematical models are often very difficult to build, due to a lack of quantitative knowledge. When qualitative knowledge is available, a knowledge-based approach can be used. The two most common knowledge-based systems are expert systems and fuzzy logic systems. According to [66], expert systems simulate the performance of human experts. An expert system consists of a knowledge base, filled with experiences of experts, and a set of precise if-then rules to apply the knowledge to problems at hand. The main difficulty with expert systems is that it is in general difficult to convert the knowledge of the experts to crisp if-then rules and that an expert system is not able to handle situations that are not

explicitly contained in the system [52]. Most expert systems are used for diagnostics only (probably due to the fact that this is the main task of most experts), but it is possible to use an expert system for prognostics as well. In [6] such a system is developed for energy conversion processes. The system is tested on real co-generation plant and did, in addition to diagnostics and prognostics, also provide the plant operator with suggestions on the course of action.

When domain knowledge cannot be fit into crisp if-then rules, but is rather vague or imprecise, a fuzzy logic system might be applicable. Fuzzy logic systems are modeled using fuzzy sets, instead of discrete values. Fuzzy logic allows the use of boolean logic and if-then rules on imprecise information. Fuzzy logic can be used as a stand alone condition monitoring solution, see e.g. [40], but is most often used in combination with other systems, for example with neural networks leading to neuro-fuzzy systems [85].

2-4-2 Data-driven methods

When quantitative or qualitative knowledge about the system is not available, but monitoring data is, statistical techniques can be used. A lot of statistical techniques are available for condition monitoring. The diagnostic problem can be seen as a statistical pattern recognition problem [27], while the prognostic problem is a prediction problem. This does lead to different techniques for both problems. For diagnostics, often basic techniques like cluster analysis [22] or statistical hypothesis tests [67] are used. An overview of the available statistical prognostic techniques is given in [65]. The most-used techniques for direct condition monitoring are regression techniques [35], Wiener process models [51], and gamma process models [75]. When the degradation signal is not observed, hazard models can be used, where instead of the degradation signal, the statistical distribution of the time of failure is modeled [34]. In these models, monitored signals are related to the instantaneous probability of failure conditional on having survived up till the current time t.

All these statistical methods assume that the patterns in the data can be described by a known model. When the patterns in the data are unknown and complex, artificial neural networks can be used, since these methods are able to describe complex structures. Neural networks can be used for both diagnostics [62] and prognostics [82]. The advantage of neural networks is that generic methods can be used for very specific circumstances. Unfortunately, an artificial neural network often remains a black box: it is unknown why an artificial neural network gives a certain output and it is found difficult to incorporate knowledge about the system in a neural network [52].

2-4-3 Hybrid methods

In hybrid methods, data-driven and model-based techniques are combined, in order to overcome problems associated with either data-driven or model-based techniques. The two most often used hybrid techniques in condition monitoring are neuro-fuzzy systems and stochastic filtering-based approaches. The most common form of neuro-fuzzy systems are fuzzy systems where the fuzzy rules are learned by a neural network. A good example can be found in [85], where a neuro-fuzzy model is used to predict bearing health. The fuzzy inference structure

is determined by expertise and the membership function are trained using a neural network. It turned out that this approach outperformed a radial basis function network.

Stochastic filtering-based approaches are methods to estimate the state of a system, e.g. the health state, using noisy observations. In these methods, data from the observed system is combined with models of the system under consideration. Stochastic filtering-based techniques can be seen as hybrid methods, but some scientists see these techniques as either model-based [3, 36, 69] or statistical [27, 52, 65]. Stochastic filtering-based approaches require a state space model of the system. In a state space model, it is assumed that future states of the system do not depend on the past, given the present state of the system (the Markov property). When the system is diagnosed using a stochastic filtering-based approach, the state space model might be used to predict future fault behavior and the remaining useful life by propagating the state space model forward in time. The main advantage of such an approach is that only one model has to be build for both diagnostics and prognostics. Three types of state space models are distinguished. This distinction is based on the states used in the model: if all states are continuous, we talk about a continuous state space model, if all states are discrete, we use the term hidden Markov model. If some states are discrete and some are continuous, the state space model is called hybrid.

The most well-known form of stochastic filtering for models with continuous states, is the celebrated Kalman filter [28]. The Kalman filter provides simple analytical expressions for the estimated state when the dynamical system is linear. The Kalman filter is often used in condition monitoring [5, 15]. For hidden Markov models, often the forward-backward algorithm [61] or Viterbi algorithm [80] is employed to obtain state probabilities or the most likely sequence of states, respectively. Using hidden Markov models for condition monitoring is extensively discussed in [4]. For hybrid models, inference is complicated in general. Most methods are computationally costly and use rigorous approximations. Filtering using generic hybrid models is not often used in condition monitoring. However, there is one specific type of hybrid systems, for which the filtering problem is extensively studied: jump Markov linear systems [17]. In models of Markov jump linear systems it is assumed that for each mode, the system can be described using a continuous-state linear state space model. It is assumed that the mode of the system evolves according to a discrete Markov chain which is independent of the continuous states.

When models of Markov jump linear systems are used in condition monitoring, we often talk about multiple-model approaches, since multiple continuous models are used: for each fault mode of the system there exists a model. Various algorithms exist for approximate multiple-model filtering. Probably, due to the possibility to accurately describe abrupt changes, multiple-model approaches are nowadays widely used for diagnostics [72, 81, 83, 84] and prognostics [37, 53, 56, 69].

2-4-4 Our preferred method

Our problem is characterized by the fact that we have monitoring data and knowledge of the system. Choosing either a model-based or a data-driven approach would exploit only one source of information: either knowledge of the system or patterns in the data. This would be inefficient use of information. Similarly, in [21] it is stated, in favor of hybrid methods, that “more complete information allows for more accurate recognition of the fault state”.

Furthermore, [79] states that both model-based and data-driven approaches for fault detection and diagnosis of track circuits face difficulties: "The difficulty with model-based approaches is that detailed system models are required, which are not easy to obtain because of the system complexity and uncertain environmental influences. The difficulty with data-based approaches is that a large and representative amount of labeled historical data is required." Hybrid approaches, might overcome weaknesses of a model-based or data-driven approach with the strengths of the other. For these reasons, we focus on methods combining both information sources: the hybrid methods. In general, neuro-fuzzy systems still require a comprehensive dataset in order to be able to generalize [57]. Since, for the track circuit condition monitoring problem little historical failure data is available, we focus on the stochastic filtering-based approaches.

We propose to use a multiple-model approach for the track circuit condition monitoring problem. This is motivated as follows. First of all, we prefer to use a state space model with both discrete an continuous states: a hybrid state space model (not to be confused with a hybrid method). We think faults in track circuits are best described using a hybrid state space model, since the track circuit case requires a combination of discrete states and continuous states. Our system can not be modeled using a single set of state equations where the state evolves continuously: for example conductive objects lead to a sudden change in system behavior, which can not be described by a continuous state space system. This is in agreement with [84], were it is argued that for diagnostic problems a stochastic hybrid system is more appropriate. Multiple-model filtering is a specific form of a stochastic filtering-based method that uses a hybrid state space model.

Second, a multiple-approach can be used for both diagnostics and prognostics. A multiple-model approach avoids developing two separate systems, which saves time and yields an integrated solution.

Third, filtering of hybrid systems is complicated in general, but for multiple-model filtering various algorithms exist for state estimation.

Lastly, a multiple-model approach allows for straightforward incorporation of information on degradation behavior and spatial influences. This prior knowledge can be incorporated in the state space models for the different fault modes. For example, if one knows that degradation behavior is exponential, a model for this degradation behavior can be developed and used in the approach. Often, grey box models are developed, where part of the model is determined by prior knowledge and part of the model is determined using system identification techniques. Prior knowledge on the transitions between modes can be incorporated as well by specifying which transitions are allowed and their probability in the transition probability matrix. The incorporation of prior knowledge has a large effect on the diagnostic performance [72] and the incorporated information is directly used for both diagnostics and prognostics.

2-5 Condition monitoring for track circuits

The literature on condition monitoring for track circuits is mainly focused on diagnostics. The literature on prognostics is scarce. For diagnostics, often neuro-fuzzy systems are used. In [63], an audio-frequency track circuit was simulated. Using a neuro-fuzzy network it was possible to detect different types of faults in the track circuits. In [12], a neuro-fuzzy system was used

as well, but the researchers tested the approach on a real track circuit in their lab, instead of on a simulation-based case study. The researchers showed that a neuro-fuzzy approach outperformed a polynomial model. In contrast to what is often encountered in practice, the researchers had eight signals available per track circuit for their diagnostic approach.

However, some studies are directed to situations where a limited amount of sensor information is available. A conceptual method for condition monitoring in networks was proposed by [79]. In this paper, the use of spatial and temporal dependencies for fault detection and diagnosis in networks is emphasized. A railway network with track circuits is used to illustrate the proposed approach. A diagnostic method for track circuits using neural networks was developed by [19]. It was shown in a simulation-based case study that excellent classification accuracy can be achieved without the use of prior knowledge. In addition, the researcher trained a neural network to estimate the intensity of the fault. The problem considered in the referred thesis, is comparable with our problem.

Not all research directed to diagnosing track circuits is based on measurements of the current flowing through track circuit relays. In [13, 20] and [45], the detection of trimming capacitor defects was studied. Those trimming capacitors (placed between the rails) can be malfunctioning. In contrast to our case, where measurements at the track side are used, the researchers used the current flowing trough the shunt of the vehicle, the short-circuit current, instead. In [20], a partial least squares regression was used to determine trimming capacitor defects. These lead to a higher resistance, which can be detected from the track circuit pattern. The signal of the short-circuit current was characterized by 17 variables, which were used as input for the partial least squares regression in order to estimate the resistance. The researchers showed that this approach works, but that a better performance is achieved using a neural network. However, the partial least squares approach is less computationally intensive and the partial least squares regression coefficients can be used as initial values for the neural network. A neural network is used in [45] as well, but in this study the neural network is combined with Dempster-Shafer classifier fusion and decision tree classifiers. This approach resulted in a correct detection rate over 99 %. The two above mentioned approaches require labeled data, which is often difficult to obtain. To overcome this problem, in [13] experts were used to classify the short-circuit current patterns, whereby the experts also reported their confidence about the classification. An independent factor analysis was performed on the data and the soft classification of the experts. When the outputs, according to the independent factor analysis for each of the four experts were combined, this resulted in a classification accuracy of 98.5 %.

When inspecting the above mentioned studies, it seems that research directed towards condition monitoring of track circuits is somewhat limited and still in its first stage. This is regrettable, since it is shown in [48] that the potential benefits of condition monitoring of track circuits are large. In [48] an availability analysis of track circuits was performed, where the researchers developed a Petri-net for track circuits and simulated different maintenance policies. The researchers showed that the availability of track circuits can be improved, compared to the current practice in Sweden, if a condition monitoring approach is employed that can detect more than 60 % of the faults.

2-6 Summary

In this chapter, the track circuit working principle and the eight track circuit faults that are distinguished in [79] were discussed. Track circuit faults obey different types of degradation behavior and some faults influence multiple track circuits. In addition, we have explained that condition-based maintenance is often advantageous over time-based maintenance, since breakdowns are avoided and maintenance costs are reduced. Condition-based maintenance requires condition monitoring, which consists of diagnostics and prognostics. Diagnostics is concerned with the classification of faults and prognostics is focused on predicting when an identified fault will lead to a failure. Furthermore, we have touched upon diagnostic and prognostic methods and we have explained why we prefer a multiple-model method. Our main reasons were: the fact that in this method monitoring data is combined with system knowledge, the ability to handle abruptly changing systems and the fact that diagnostics and prognostics are integrated. Lastly, an overview was given of existing literature for condition monitoring of track circuits and it was concluded that the literature on prognostics for track circuits is scarce.

Chapter 3

Multiple-model state estimation and prediction

In the previous chapter we have motivated the choice for a multiple-model approach. In this chapter, we discuss the multiple-model methodology and make choices with respect to the techniques used for state estimation and prediction. We start by defining the underlying Markov jump linear system that is assumed in a multiple-model approach in Section 3-1. Next, we outline the different views that exist on Markov jump linear systems in Section 3-2. Furthermore, we explain the differences between the three types of multiple-model approaches that exist and select the best suited one (Section 3-3). Next, we outline state estimation in a multiple-model approach and evaluate the available approximate state estimation algorithms (Section 3-4). Lastly, we report how predictions for the remaining useful life can be made (Section 3-5). In Section 3-6, we summarize the contents of this chapter.

3-1 Markov jump linear systems

In a multiple-model approach it is assumed that for each mode, the system can be described using a (possibly different) continuous linear state space model. In general, it is assumed that the mode follows a discrete Markov chain. Such a system is called a Markov jump linear system. In a Markov jump linear system, it is assumed that the system can be described according to:

$$x_{t+1} = A(\theta_t)x_t + B(\theta_t)u_t + v_t \tag{3-1}$$

$$y_t = C(\theta_t)x_t + D(\theta_t)u_t + w_t, \tag{3-2}$$

where

$$E[v_t v_t^T] = Q(\theta_t) \tag{3-3}$$

$$E[w_t w_t^T] = R(\theta_t) \tag{3-4}$$

and θ_t follows a discrete Markov chain. Here x_t denotes the state, u_t the input, w_t is process noise, v_t measurement noise and A, B, C, D, Q, R are matrices that are dependent on the discrete mode of the system θ_t. With t we denote the current time step. From the above formulation it becomes clear that when the mode is known, the system is described using a linear discrete-time continuous-state state space model. In condition monitoring, Markov jump linear systems are used to describe a system under different fault modes. For each possible value of θ, the system operates according to a different model, i.e. different matrices A, B, C, D, Q and R.

3-2 Views

Two important views exist on Markov jump linear systems [17]: the multiple-model point of view and the analytical point of view. Below, we briefly explain the difference between the two views:

- In the multiple-model point of view, the focus is laid on determining the actual mode of the system and filtering and predicting using the model for that mode [24]. For example, a model is developed for healthy system behavior and another model is made to describe the behavior when a system suffers from an actuator fault. Multiple-model filters estimate what model is currently active. This is done by inspecting what model explains the data best: if the model of an actuator fault explains the data very well, while the model for healthy behavior does not, one expects that the section currently suffers from a faulty actuator. To find what model explains the data best, an observer is used for each fault mode, which returns estimates of the continuous states and the likelihood of the data given the model. Using the likelihood of the data given each mode, the active mode of the system can be estimated by applying Bayes rule. A multiple-model approach thus requires a model (matrices A, B, C, D, Q and R) for each mode. In the condition monitoring community often the multiple-model point of view is taken [24, 37, 72]. Possibly, this originates from the fact that diagnostics is often concerned with hypothesis testing. Determining which model explains the data best can be seen as a likelihood ratio test, which is a traditional hypothesis testing technique.
- In the analytical point of view, the systems are approached as ordinary state space systems with augmented states $\{x_t, \theta_t\}$, where x_t are the continuous states and θ_t is the discrete state representing the mode [17]. Determining the correct model is not more important than estimating the continuous states. From an analytical point of view, there do not exist multiple models: there exists only one model, which has a hybrid state.

These different point of views lead to different approaches. Most methods are derived based one of the two point of views and cannot be easily explained with the other. In this thesis, we adopt the multiple-model point of view, since it is suited to condition monitoring and we think it is the most intuitive interpretation. This facilitates the incorporation of prior knowledge into our approach. However, when discussing state-estimation algorithms, we sometimes adopt the analytical point of view, in order to explain approaches based on that point of view.

3-3 Three types of multiple-model approaches

Three types of multiple-model approaches exist [58]: static multiple-model approaches (the first generation) [38], dynamic fixed structure multiple-model approaches (second generation) [1, 7, 11] and dynamic variable structure multiple-model approaches (third generation) [33]. We first outline the difference between the static and dynamic approaches and then discuss the difference between the fixed structure and variable structure dynamic multiple-model approaches.

3-3-1 Static versus dynamic multiple-model approaches

The first distinction is between static and dynamic multiple-model approaches. In a static multiple-model approach it is assumed that the observers that are used to detect what model explains the data best, run in parallel without interaction. In such an approach, it is assumed that only one model is active the entire time the system is observed: the system cannot switch between modes (θ_t is constant for the entire sequence, but unknown). In dynamic multiple-model approaches it is assumed that the system can switch between modes: the section can be healthy up to time t and suffer from a faulty actuator afterwards. Switching between modes is modeled using a Markov chain. Where static multiple-model approaches are aimed at determining an unknown structure or parameter, dynamic multiple-model approaches are aimed at the detecting sudden changes.

We propose to use a dynamic multiple-model approach, since a fault is a structural change of the system. If we would use a static multiple-model approach, we would implicitly test whether according to the data a fault has always been present or has never been present, while in a dynamic multiple-model approach we test whether a change occurred in the system. This describes our situation best: track circuits can be healthy first and later suffer from a fault. We want to detect whether such a switch has occurred.

3-3-2 Fixed versus variable structure dynamic multiple-model approaches

Within the dynamic multiple-model approaches, two forms are distinguished: fixed structure approaches (second generation) and variable structure approaches (third generation). In the fixed structure approaches, a fixed set of models is used for filtering, where it is assumed that the set of models is constant over the entire time the system is observed. These approaches experience substantial difficulties when a large number of models is required to describe the system. First of all, using more models increases the computational burden. Second, it might also lead to worse filtering performance, due to competition from unnecessary models. To solve these problems, a variable-structure approach can be used. In such an approach the model set is not constant over time and the models that are included in the set are based on on-line information [33]. In practice, only models are included that are possible given the current mode of the system. This means that the model set depends on the current state of the system and therefore can be varying.

For our problem, the number of modes that has to be distinguished is relatively small. Therefore, the expected gains of using a variable structure are small. Furthermore, the most important diagnostic task for our approach is to detect a change from the healthy mode to one

of the faulty modes. This means that when the system currently exhibits healthy behavior, switches to all fault modes are possible and models for these fault modes should therefore be included in the model set, i.e. a fixed model set for the majority of the time. In this perspective, the gains of a variable structure multiple-model approach are small as well. We therefore prefer a fixed structure. To summarize, we select a dynamic fixed-structure approach, also known as a second generation approach.

3-4 State estimation

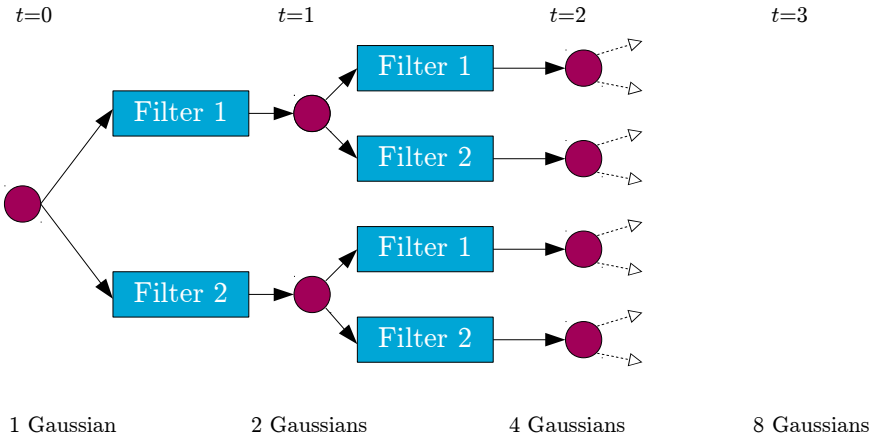
In this section, we discuss state estimation in a multiple-model approach, which is used for diagnostics. State estimation is used to estimate the mode (the discrete health state θ_t) of the system. In this section, we describe why exact state estimation is intractable in general and what approximations can be used to overcome this problem in Section 3-4-1. Thereafter, we discuss the interacting multiple-model approach in somewhat more detail in Section 3-4-2.

3-4-1 Algorithms

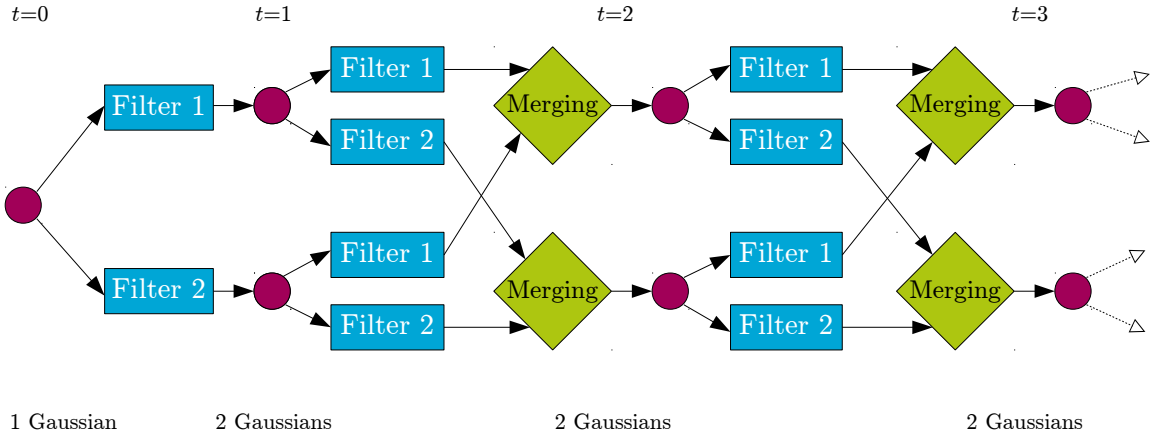
In a multiple-model approach, we determine what model explains the data best using an observer for each fault mode. These observers return estimates of the continuous states x_t and the likelihood of the data given the model for that mode $p\left(y_t|\theta_t = j\right)$. Using the likelihood of the data given each mode, the active mode of the system can be estimated by applying Bayes rule. Since the models for each mode are linear state space models, Kalman filters are used. In general, it is intractable to calculate the optimal state estimate for a dynamic multiple-model configuration. This intractability is caused by the fact that the number of hypotheses that has to be considered in every filtering step increases exponentially with time.

We illustrate this with an example. Assume that there exist two modes: mode A and mode B. We want to estimate the state of the system at $t = 1, 2$, where t is defined as the time step. We know that at $t = 0$ the system is in mode A. At $t = 1$ two hypotheses exist: the hypothesis that the system stayed in A and the hypothesis that the system switched at $t = 1$ to B. At $t = 2$ we have four hypotheses: the system stayed in mode A at $t = 1$ and stayed in mode A at $t = 2$ as well, the system stayed in mode A at $t = 1$ and switched to mode B at $t = 2$, the system stayed in mode A at $t = 1$ and switched to mode B at $t = 2$, the system switched to mode B at $t = 1$ and stayed in mode B at $t = 2$ and lastly the hypothesis that the system switched to mode B at $t = 1$ and switched back to mode A at $t = 2$. In all exact solution that are known to date, all these hypotheses have to be considered. For multiple linear models with additive Gaussian white noise, this leads to continuous-state estimates that consist of a mixture of M^t Gaussians [1]. Here M is the number of models and t is the current time step. We have illustrated this in Figure 3-1a. The cyan circles represent a Gaussian and the blue boxes represent the filters running in parallel at each time-step.

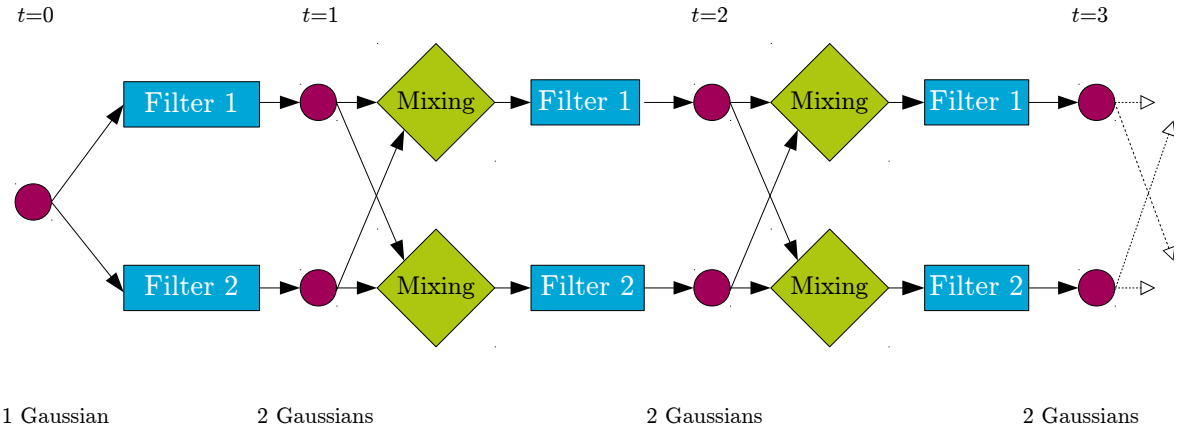
To make estimation feasible, one has to use an approximation. Various approximation algorithms have been developed for multiple-model filtering. From a multiple-model viewpoint two types of approximations can be distinguished: pruning strategies and merging strategies [24]. Pruning strategies reduce the number of hypotheses by cutting off unlikely branches. Merging strategies do not cut off certain hypotheses, but merge hypotheses that are similar.



(a) Exact.



(b) Generalized pseudo Bayes 2 algorithm.



(c) Interacting multiple-model algorithm.

Figure 3-1: Filtering using a multiple-model approach with $M = 2$ models [42].

Next to approximations derived from a multiple-model point of view, there exist algorithms that are derived from an analytical point of view. We have summarized the most important methods below.

- Pruning strategies
  - In the N best strategy, originally called "detection-estimation scheme", only the B most likely sequences are kept. At each time step, filtering is performed for each model using state estimates of all the N most likely hypotheses [73]. This results in MN new state estimates. Only the N hypotheses with the highest likelihood are contained, the rest is pruned. The number of best models to retain N can be constant over time, but can also be time-varying [58].
  - With the Viterbi strategy [58], for each of the M models the most likely sequence leading to that model is maintained. The rest is pruned. This means that for each time step, a hypothesis exist whereby the system is now in one of the M modes. This ensures that the best model for each filter is always maintained. However, often not the M most likely hypotheses are maintained, since, for example, the second best hypothesis ending at model 1 might be more likely than the most likely hypothesis ending at model 2.
- Merging strategies
  - With the generalized pseudo Bayes k algorithm, Gaussian hypotheses are collapsed that had a different history k or more time steps ago, but have the same history after k time steps ago [1, 11][1]. Due to this merging, the generalized pseudo Bayes algorithm requires M^k continuous-time filters running in parallel. The generalized pseudo Bayes k algorithm is shown for $k = 2, M = 2$ in Figure 3-1b.
  - In the interacting multiple-model algorithm, the obtained hypotheses are mixed before processing the continuous-state filters, whereas in the generalized pseudo Bayes algorithm hypotheses are merged after processing of the continuous-state filters [7]. This allows the interacting multiple-model algorithm to use only M filters in parallel. The interacting multiple-model algorithm is shown for $M = 2$ in Figure 3-1c.
- Methods derived from an analytical point of view
  - The linear minimum mean square error estimator estimates a constructed state $z_t = (1_{\theta_t=i}^\top, x_t 1_{\theta_t=i}^\top)^\top$ where x_t is the continuous state, θ_t is the mode and 1 stands for the Dirac measure [16]. An optimal linear estimate of z_t is obtained and this is used to calculate estimates of x_t and $1_{\theta_t=i}$. The filter reduces to the Kalman filter when only one mode is considered.

For all algorithms, little is known about their theoretical performance. Only for the linear minimum mean square estimator we know that this estimator is the optimal linear estimator, but it remains unknown whether the optimal linear estimator is close to the optimal estimator.

[1] In [11] it is stated that the generalized pseudo Bayes 2 estimator is optimal, but the algorithm is suboptimal as is shown in [74].

Therefore, for the choice of the algorithm, we have to rely on simulation-based case studies and known characteristics, such as the computational complexity of the algorithm.

Different views exist on whether merging or pruning strategies are better for multiple-model filtering. In [24], it is stated that with merging "all discrete information of the history is lost" and that therefore "merging is less useful for fault detection and isolation than pruning." On the other hand, it is stated in [58] that "merging outperforms pruning and selection particularly when the true model differs significantly from those assumed." We think the first argument is not applicable, since we are not interested in past health states under consideration, but only in the current health state. The argument made in [58] is more relevant, since in most condition monitoring applications the exact form of degradation behavior is unknown and the models used are approximations that deviate from the 'true' model. Therefore we prefer a merging strategy over a pruning strategy.

In [7], the generalized pseudo Bayes 1 and 2 algorithm and the interacting multiple-model algorithm are compared. The authors conclude that "the interacting multiple-model algorithm performs almost as well as the generalized pseudo Bayes 2 algorithm, while its computational load is about that of generalized pseudo Bayes 1." Due to this nice trade-off between accuracy and computational complexity, the interacting multiple-model algorithm is widely used nowadays [24].

In [16], the interacting multiple-model algorithm was compared with the linear minimum mean square error estimator for additive noise using 15 simulation-based cases. In ten cases the filtering performance was about equal, in five cases the interacting multiple-model algorithm outperformed the linear minimum mean squared error estimator. However, in these cases the difference was very small.

We propose to use the interacting multiple-model algorithm, since its outperforms the generalized pseudo Bayes 2 algorithm in terms of computational complexity. Furthermore, in contrast to the linear minimum mean square error estimator, the interacting multiple-model algorithm is widely used for condition monitoring and MATLAB implementations of the algorithm are available [25].

3-4-2 The interacting multiple-model algorithm

To explain the working principle of the interacting multiple-model algorithm, we use the schematic overview presented in Figure 3-2. We assume that the outputs of the system y_t are observed. Observing the inputs u_t is optional and only required when the models used incorporate inputs. The goal is to determine

- the probability that each mode is active $\mu_t^i = p\left(\theta_t = i | y_{1:t}\right)$
- the continuous state estimates $\bar{x}_t^i$ according to each model i with covariance P_t^i
- the overall continuous state estimates $\bar{x}_t$ with covariance P_t

With the superscript $^+$ we denote the mixed state estimate as input for a Kalman filter. With π_{ij} we denote the transition probability from mode i to mode j.

From this figure we can see that one iteration of the interacting multiple-model algorithm consists of a number of steps. The steps for the iteration at t are:

1. First the previous obtained state estimates for all filters $i = 1, ..., M$ are mixed based on the assumption that a particular mode j is now active, leading to mixed estimates of the previous state which are used as inputs for the Kalman filters. The mixed estimates are calculated using the mixing probabilities $\mu_t^{i,j}$, which denote the probability that the system was in mode i in time step $t-1$ and is now in mode j. The mixing probabilities are calculated as

$$\bar{c}_j = \sum_{i=1}^{M} \pi_{ij} \mu_{t-1}^i, \tag{3-5}$$

$$\mu_t^{i,j} = \frac{1}{\bar{c}_j} \pi_{ij} \mu_{t-1}^i. \tag{3-6}$$

   Using these mixing probabilities, the mixed estimates and covariance of the previous time step for filter j are determined:

$$\bar{x}_{t-1}^{j+} = \sum_{i=1}^{M} \mu_t^{i,j} \bar{x}_{t-1}^i, \tag{3-7}$$

$$P_{t-1}^{j+} = \sum_{i=1}^{M} \mu_t^{i,j} \left(P_{t-1}^{i-} + \left(\bar{x}_{t-1}^i - \bar{x}_{t-1}^{j+} \right) \left(\bar{x}_{t-1}^i - \bar{x}_{t-1}^{j+} \right)^\top \right). \tag{3-8}$$

   This is denoted with 'interacting/mixing' in Figure 3-2.

2. Based on the mixed estimates of the previous state $\bar{x}_{t-1}^{j+}$ with covariance P_{t-1}^{j+} and optionally the system inputs u_t, filtering is performed using each Kalman filter j, which leads to estimates of the current state $\bar{x}_t^j$ with covariance P_t^j (the blue boxes denoted with 'Kalman filter' in Figure 3-2). Next to the state estimates, the likelihood of the measurement y_t given the model j is calculated. We denote this with $L_t^j = p\left(y_t | \theta_t = j\right)$.

3. Next, the probability that a mode is currently active is updated, based on the probability prior to filtering of being in mode j, denoted with $\bar{c}_j$, and the likelihood of the new observation L_t^j according to model j:

$$c = \sum_{i=1}^{M} L_t^i \bar{c}_i, \tag{3-9}$$

$$\mu_t^j = \frac{1}{c} L_t^j \bar{c}_j. \tag{3-10}$$

   Calculation of these mode probabilities is done in the cyan box denoted with 'mode probability update and mixing probability calculation' in Figure 3-2.

4. Using the probabilities of a particular fault being active μ_t^j, a decision can be taken on the course of action. In the figure this is denoted with 'fault decision'.

5. Furthermore, an overall estimate of the continuous state can be obtained, using:

$$\bar{x}_t = \sum_{i=1}^{M} \mu_t^i \bar{x}_t^i, \tag{3-11}$$

$$P_t = \sum_{i=1}^{M} \mu_t^i \left(P_t^i + \left(\bar{x}_t^i - \bar{x}_t \right) \left(\bar{x}_t^i - \bar{x}_t \right)^\top \right). \tag{3-12}$$

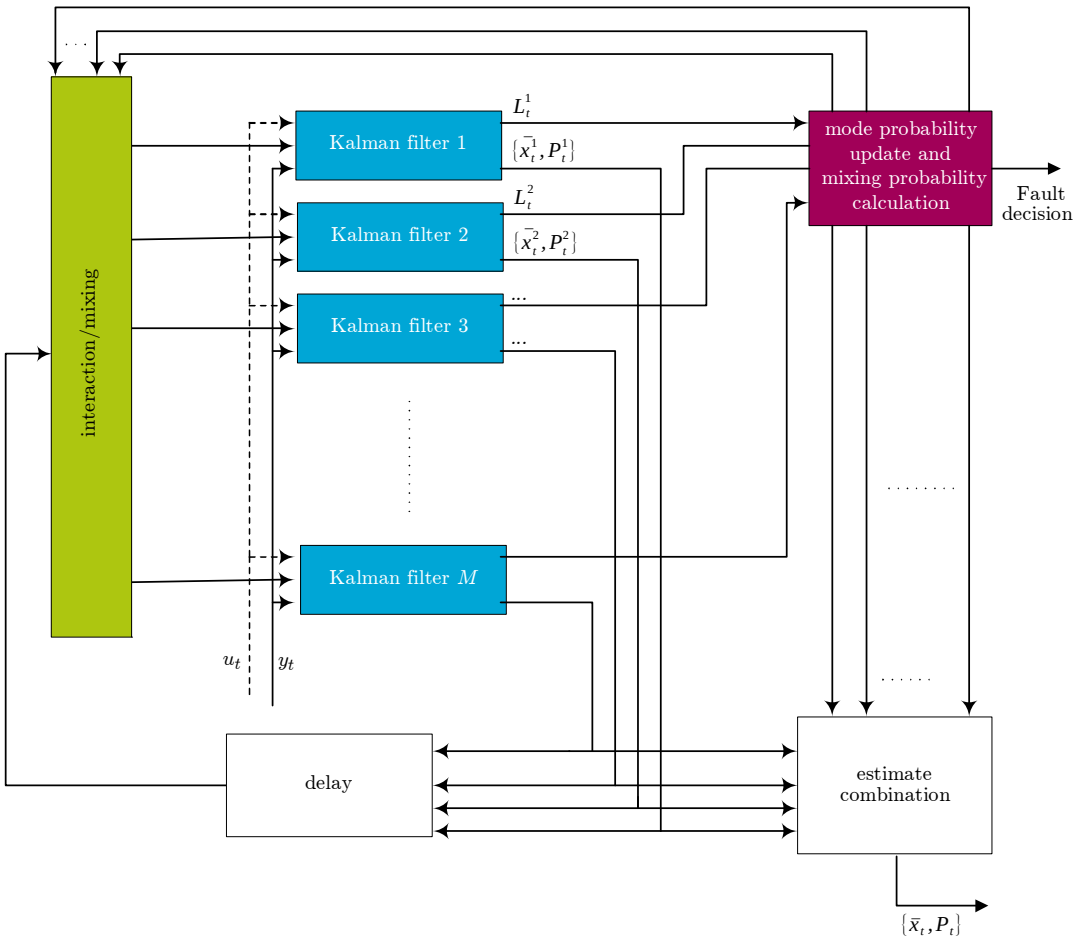


Figure 3-2: The interacting multiple-model concept [81].

This is denoted with 'estimate combination' in Figure 3-2.

6. At the end of each time step, the cycle start again. However, now the obtained estimates at time step t have become $t-1$ in the figure. We have denoted this with the delay in Figure 3-2.

3-5 Predicting the remaining useful life

Once the active fault mode θ_t is determined and the continuous states $\{\bar{x}_t^j, P_t^j\}$ of each model are obtained, one can make a prediction for the remaining useful life. To estimate the remaining useful life, the time the degradation signal crosses the failure threshold has to be predicted (see Section 2-3). The most straight forward way of predicting the remaining useful life is by propagating the state space model of the detected fault forward in time, see e.g. [56].

The main problem with this approach is that uncertainty in classification, state estimation, future inputs and process and measurement noise is not taken into account. These uncertain-

ties will not only lead to uncertainty around the point prediction for the remaining useful life, but might influence the point prediction itself as well. This can be illustrated by an example. Assume that in mode A the system has an expected remaining useful life of two years and in mode B the expected remaining useful life is a week. The probability of mode A is 55 % and the probability of mode B is 45 % according to the state estimation algorithm. When only the most likely mode is taken into account, the expected remaining useful life will be two years. In contrast, if uncertainty around the mode is considered and both modes with their respective probability are taken into account, the expected remaining useful life will be shorter than two years. Next to the fact that uncertainty influences point predictions, information on the uncertainty of the prediction is important for the decision maker as well: he has to decide whether the object has to be replaced.

In order to take uncertainty into account, one could choose a Monte Carlo approach [36]. In a Monte Carlo approach, instead of one prediction, multiple predictions, called runs, are performed. For each run, different initial conditions and noise realizations are used. By inspecting the spread in the calculated remaining useful life of all runs, one can determine the uncertainty in the prediction. Based on all the runs, the probability density function and cumulative density function can be estimated using a histogram or kernel density estimation [47, 60]. In practice, for each run a model is drawn according to the estimated probabilities that each model is active, e.g. if the probability of model A is 90 % and the probability of model B is 10 %, one draws on average model A nine out of ten times and model B once out of ten. Afterwards, for each run a state is drawn from a Gaussian distribution with as mean the estimated state $\bar{x}_t^j$ for the drawn model j and as covariance the estimated covariance P_t^j according to that model. Next, future model inputs $u_{t+1}, ..., u_T$ are drawn based on historical data and random process noise $w_{t+1}, ..., w_T$ and measurement noise $v_{t+1}, ..., v_T$ is generated. Using all these randomly drawn quantities, a prediction is made and the time until failure is recorded. Based on the remaining useful life predictions of all runs, the remaining useful life distribution is estimated and a point prediction can be obtained.

It is possible to validate the uncertainty around the predictions made by the Monte Carlo approach by comparing the number of observations falling in the confidence interval and the number of observations that should fall in the confidence interval. Sometimes, a discrepancy exists between the nominal value of the confidence interval (for example 95 %) and the true value exist (for example 92 % of the observations falls within the 95 % confidence interval). By changing the amount of process and measurement noise, it is possible to tune the predictions so that the true value of the confidence interval approaches the nominal value. In condition monitoring applications, sometimes an outer-correction loop is used for this purpose, whereby the process and measurement noise matrices Q and R are adapted on-line, based on short-term forecasting errors [44, 69].

In a Monte Carlo approach, only known uncertainties are taken into account. This gives a better estimate of the remaining useful life than making a single prediction without taking uncertainty into account at all. Predictions get closer to reality if more effects are taken into account. However, it is possible that unmodeled uncertainties or incorrectly modeled uncertainties have a large effect on the remaining useful life. A Monte Carlo approach does not give information on these effects, which has as disadvantage that information on the modeled uncertainty can be mistaken for information about the true uncertainty. Therefore, outcomes of Monte Carlo approaches should always be handled with care.

Another difficulty with a Monte Carlo approach is that enough runs have to be generated to obtain a good approximation of the probability density function. To test whether enoughs runs are used, one can perform a Monte Carlo approach using more runs. Once adding more runs does not have a (large) effect on the outcomes, it is assumed that the Monte Carlo method has converged. We have performed such an analysis for our simulation-based case study in Section 2 of Appendix B.

We believe that giving information about the uncertainty of the predicted remaining useful life is very important for the decision maker. Providing an approximation of the uncertainty around the predictions is better than providing no uncertainty information at all. For example, if the estimated remaining useful life of a track circuit is one year, an uncertainty of one year around this estimates leads to other decisions than an uncertainty of one week around the estimate. In the first situation, often the component would be replaced immediately, while in the latter case maintenance can be scheduled for a few months ahead. Furthermore, uncertainty in classification, state estimation, future inputs and process and measurement noise might affect the point prediction of the remaining useful life. Therefore we think these uncertainties should be taken into account when making predictions. For these two reasons, we select a Monte Carlo approach for prognostics.

3-6 Summary

In this chapter, we have studied the multiple-model methodology. We have discussed the difference between static and dynamic approaches and the difference between fixed and variable structure dynamic multiple-model approaches. A fixed structure dynamic multiple-model approach was selected. Furthermore, we have discussed how state estimation is performed in a multiple-model setting and outlined the most important algorithms that are available. The interacting multiple-model algorithm is preferred, because of its excellent performance in combination with a low computational cost. For predicting the remaining useful life, two options were considered: point predictions and a Monte Carlo approach. A Monte Carlo approach was chosen, since using this method known uncertainties can be taken into account.

Chapter 4

A multiple-model approach applied to track circuits

In the previous chapter, we have studied multiple-model state estimation and prediction. In this chapter, we discuss how a multiple-model approach can be used for track circuit condition monitoring. After analyzing the track circuit condition monitoring problem in Section 4-1, we describe how the measurements can be preprocessed to make them suitable for condition monitoring in Section 4-2. Next, we derive models for each health state of the system and describe what transitions between health states are allowed (Section 4-3). Lastly, we describe which practical problems are encountered when using a multiple-model approach for track circuit condition monitoring and propose solutions in Section 4-4. The main conclusions of this chapter our outlined in Section 4-5.

4-1 Problem analysis

The condition monitoring approach has to determine the health state of the track circuit and to predict the remaining useful life. In [79], eight faults are distinguished, which are presented in Table 2-1. We deviate slightly from the presented classification and do not consider lightweight trains ($\theta = 2$) and ballast variation ($\theta = 8$) as faults, since these influences are not related to condition of the track circuit itself. However, it is still required to incorporate their influence into our condition monitoring approach, e.g. ballast variation should not be confused with ballast degradation. Furthermore in order to limit the scope of this thesis, we focus on faults that could lead to a false positive train detection. Fortunately, the presented solution can be easily applied to distinguish between healthy behavior and rail contamination, which is the only fault that could lead to a false clear-section signal. Since currently no information is available about the interaction between faults, we focus in this thesis on the effect of one fault at the same time. In theory, the proposed approach could be extended in the future to situations where multiple faults are present. In that case, each possible combination of health states should be seen as a distinct health state and models should be developed for each health state combination.

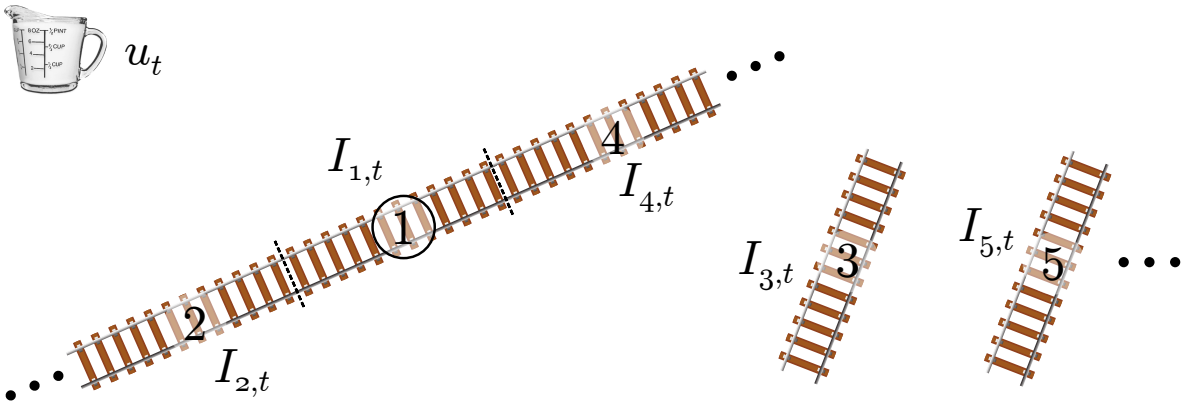


Figure 4-1: Schematic overview of the signals used.

We propose a condition monitoring approach that monitors a single track circuit. Monitoring multiple track circuits using one system would lead to a combinatorial explosion of the number of health states distinguished in the system. For example, we would end up with health states like 'track circuit 1 suffers from a mechanical defect and track circuit 2 suffers from ballast degradation.' Therefore, we choose to build a system that monitors one track circuit. However, we still incorporate information of other track circuits in our approach, but we only diagnose one track circuit and make predictions for that track circuit only. We denote the track circuit that we want to monitor with a 1 in Figure 4-1. For our condition monitoring approach, we use two types of monitoring data: track circuit measurements $I_{j,t}$ for each section j and rain sensor data u_t. The track circuit measurements are measurements of the current flowing through the relay of the track circuit (see Section 2-1). The rain sensor outputs a signal that is proportional to the amount of rainfall. The use of these signals is advantageous, since for a lot of track circuits, measurements of the current flowing through the relays is already stored and available for condition monitoring. Furthermore, industrial rain sensors are relatively cheap ($< €\ 500$) and can be easily installed. Therefore, the proposed approach leads to low installation costs. We have depicted the signals used schematically in Figure 4-1. We assume that a rain sensor is located in each region of ± 2 km^2.

In order to distinguish between faults, it is suggested to make use of the spatial dependencies in the railway network [79]. Using these dependencies, it is possible to distinguish between faults that influence only one section and faults influencing all track circuits on the same track. Furthermore, it allows us to distinguish between ballast degradation and ballast variation. Next to current measurements of the monitored track circuit $I_{1,t}$, we therefore use measurements of nearby track circuits as well. We make a distinction between current measurements of other track circuits on the same track $I_{2,t}, I_{4,t}, ...$ and of nearby track circuits on other tracks $I_{3,t}, I_{5,t}, ...$ We only use clear-section measurements, since we focus on faults that could lead to a false positive train detection and those can be detected on a clear section only. The signal is sampled 50 times a day equidistantly. When the section is occupied at the time of sampling, we suggest that a measurement just before the section got occupied is used instead, e.g. a measurement of 40 seconds earlier. When a fault occurs, the current will be lower than expected ($I_{1,t} < \alpha_2$). A fault leads to a failure when $I_{1,t} < \gamma_2$ (see Section 2-1-2).

The measurements of nearby track circuits cannot be used directly for condition monitoring. Developing models that relate all track circuit measurements directly, would lead to a

combinatorial explosion of the number of models, since we need to create a model for each possible combination of health states. For example, we would end up with a model specific for a situation where track circuit 1 suffers from a mechanical defect, track circuit 2 suffers from ballast degradation, track circuit 3 is healthy and track circuit 4 suffers from a conductive object. This would lead to an enormous amount of models, which would make our condition monitoring approach slow and might even deteriorate the filtering performance [33].

To overcome this problem, we aim at developing models where the relation between the inputs u_t and outputs y_t can be accurately described given the health state of the monitored track circuit $\theta_{1,t}$ only. This means the active model m_t describing the relation between u_t and y_t may not depend on $\theta_{2,t}, ..., \theta_{J,t}$ given $\theta_{1,t}$, i.e. $m_t = f(\theta_{1,t})$. This constrains the number of distinguished modes to the number of possible discrete values of $\theta_{1,t}$. To accurately describe the relation between the inputs u_t and y_t using these models, it is required that, given the health state of the monitored track circuit $\theta_{1,t}$, the signals u_t and y_t do not depend on $\theta_{2,t}, \theta_{3,t}...$, either.

The signals u_t consist of rain sensor measurements, which do not depend on any health state. Therefore no preprocessing is required for u_t. However, the measured currents $I_{j,t}$ do depend on the health states $\theta_{j,t}$. Therefore some transformations are required before these signals can be used for multiple-model filtering. The information that is required for accurate fault diagnosis and prognosis is:

- the relation between the current of the monitored track circuit and currents of nearby track circuits on the same track
- the relation between the current of the monitored track circuit and currents of nearby track circuits located at other tracks

We propose to use three signals to capture these relations:

- $y_{1,t}$ - the current flowing through the monitored track circuit
- $y_{2,t}$ - the current of nearby track circuits on the same track
- $y_{3,t}$ - the current of nearby track circuits on other track

4-2 Preprocessing

The signals $y_{2,t}$ and $y_{3,t}$ have to be constructed from $I_{2,t}, I_{4,t}....$ and $I_{3,t}, I_{5,t}, ...$, respectively, i.e.

$$y_{2,t} = f(I_{2,t}, I_{4,t}....) \tag{4-1}$$
$$y_{3,t} = f(I_{3,t}, I_{5,t}....). \tag{4-2}$$

We call this the preprocessing stage. In this stage the main goal is to create signals $y_{2,t}$ and $y_{3,t}$ that are not affected by faults influencing a single nearby track circuit, as described earlier. We propose two techniques to achieve this.

First of all, we suggest to exclude measurements of track circuits suffering from a fault that influences only that particular track circuit. If a fault influencing a single section is detected at track circuit 2 in time step $t-1$, we do not use that measurement for condition monitoring of track circuit 1 (the track circuit to monitor) in time step t. This is possible since, in

practice, condition monitoring will be performed for multiple track circuits and therefore the health state of nearby track circuits at the previous time step is known. If a measurement is excluded, the signal $y_{2,t}$ or $y_{3,t}$ will be based on the other track circuit measurements. This is possible, since the prescribed currents of all track circuits are equal[1] [54]. When almost all measurements to calculate $y_{2,t}$ or $y_{3,t}$ have to be excluded (since all nearby track circuits suffer from a fault influencing a single track circuit), we suggest to extend the range of considered track circuits to ensure that a minimum amount of measurements is used to calculate $y_{2,t}$ or $y_{3,t}$.

Second, to make the system resistant to faults that are not (yet) detected, we propose to use the median to combine the measurements, instead of the widely-used mean. The median is very resistant to outliers and therefore the influence of an undetected fault, which is an outlier, is minimized. We have also considered the weighted median as an option, whereby measurements of nearby track circuits get more weight than track circuits far away. According to [71], weighting is a bad idea for nonlinear combination schemes. In general, the errors introduced by estimating the weights nullify the performance increase gained by using weights.

The optimal number of nearby track circuits to take into account is situation-specific: in some situations, the number of track circuits located close to the monitored one is large, while in other situations measurements are only available for one nearby track circuit. In general, increasing the number of track circuits considered leads to an increased robustness against undetected faults. On the other hand, environmental influences are local and using measurements of track circuits far away from the track circuit to be diagnosed, might deteriorate the monitoring performance. Finding the optimal number of track circuits requires extensive simulation work. To limit the scope of this thesis, we do not focus on finding this optimum.

As a result of our preprocessing phase, we use four signals in our multiple-model approach:

- u_t - rain sensor measurements
- $y_{1,t}$ - the current flowing through the relay of the track circuit to monitor
- $y_{2,t}$ - median of currents flowing through relays of nearby track circuits on the same track
- $y_{3,t}$ - median of currents flowing through relays of nearby track circuits on different tracks

We see $y_{1,t}$, $y_{2,t}$ and $y_{3,t}$ as outputs and u_t as inputs for the models used in the multiple-model approach, since the amount of rainfall causes a change in the observed current. With this notation there exists a clear distinction between raw track circuit measurements I_t and preprocessed track circuit measurements used for multiple-model filtering y_t.

4-3 Modeling

The basis of our condition monitoring program is the multiple-model framework, as described in Chapter 3. For this multiple-model framework, linear state space models for each mode of the system have to be developed and one has to specify which transitions between modes are allowed. In this section, we define these models and specify the allowed transitions.

[1] Except a few located near Best.

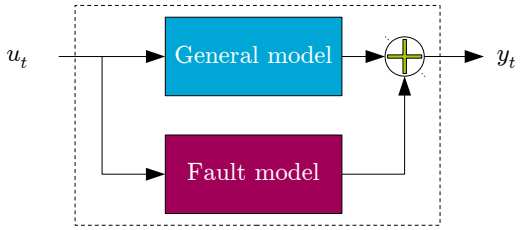


Figure 4-2: Structure of the state space models.

We split the development of the state space models into two stages, whereby we follow [19] and assume faults have an additive effect on the normal operating behavior. This means we develop a model for track circuit measurements in general and a set of fault models (see Figure 4-2). The general model is common for all modes of the system, while the fault model is mode-specific. The general model has to be combined with the appropriate fault model leading to one model for every mode of the system. The linear state space models have to relate the inputs u_t to the outputs y_t.

We start by defining a model for track circuit measurements in general in Section 4-3-1 and describe how to model each type of fault in Section 4-3-2. Lastly, we describe the transitions between the modes in Section 4-3-3.

4-3-1 A general model for track circuit measurements

In [19], the characteristics of track circuit measurements are studied. We build further on this work as basis for deriving our models.

For clear sections, in a fault free situation, we observe a high current $I_{j,t}$ flowing through the relay. The nominal current slightly differs for each track circuit[2] and therefore for each output $y_{s,t}$. We denote the nominal current of each output with μ_s.

The track circuit currents vary over time. The amount of variation is different for each signal $y_{s,t}$. Part of the variation is caused by the wetness of the ballast, another part depends on the time of the day, and yet another part of the variation is caused by unknown factors.

In [19], the wetness of the ballast β_t of all sections in the region is described using a first-order autoregressive process:

$$\beta_{t+1} = g\beta_t + u_t + \zeta_t. \tag{4-3}$$

Here $u_t + \zeta_t$ represents the rain at time t. The autoregressive coefficient is denoted with g, where $0 \leq g < 1$ to ensure a diminishing effect of rainfall. The part u_t is the amount of rain that is measured by a rain sensor. Furthermore, ζ_t represents the part of the rain which is not measured by the sensor. We model ζ_t as independent identically distributed Gaussian noise with mean zero and variance σ_ζ^2. The mean and variance of the distribution are identified using system identification. The influence of wetness of the ballast on a particular output $y_{s,t}$ is denoted with the coefficient a_s.

[2]The nominal current level prescribed in the installation manual is equal for all track circuits (except track circuits located near Best), but due to manual tuning and location-specific circumstances the actual nominal current differs a little bit [54].

The current flowing through the relays also depends on the time of the day. A deterministic sinusoidal signal with a period of 24 hours can approximate this behavior well. We denote it with λ_t and denote the influence of periodic effects on the output $y_{s,t}$ with b_s. We denote the frequency of the periodic effects with ω.

Other unknown causes of short-term variation are denoted with $\eta_{s,t}$ and ϵ_t. Here $\eta_{s,t}$ includes unknown causes of short-term variation influencing either measurements of nearby track circuits on the same track $y_{2,t}$, nearby track circuits on other tracks $y_{3,t}$ or the track circuit to monitor $y_{1,t}$. The term ϵ_t represents unknown causes of short-term variation that influence all outputs $y_{s,t}$. Both sources are modeled as independent and identically distributed Gaussian noise with mean zero and variance $\sigma^2_{\eta,s}$ and σ^2_{ϵ} respectively. We denote the influence of ϵ_t on each output $y_{s,t}$ with c_s.

Next to short-term variation, we also observe long-term variation. Probably, this is for a large part caused by seasonal influences [19]. We model both output-specific long-term variation causes $\chi_{s,t}$ and long-term causes affecting all outputs ξ_t as random walk processes:

$$\chi_{s,t+1} = \chi_{s,t} + \nu_{s,t} \tag{4-4}$$

$$\xi_{t+1} = \xi_t + \kappa_t. \tag{4-5}$$

Here, $\nu_{s,t}$ and κ_t are modeled as independent and identically distributed Gaussian noise. The influence of ξ_t on a specific output s is modeled with d_s.

The output, subject to all the described influences, can be described using:

$$y_{s,t} = \mu_s + a_s\beta_t + b_s\lambda_t + \eta_{s,t} + c_s\epsilon_t + \chi_{s,t} + d_s\xi_t. \tag{4-6}$$

These descriptions can be put into discrete-time state space form. The relations result in a linear system and can therefore be described using a discrete-time linear state space system:

$$x_{t+1} = Ax_t + Bu_t + v_t \tag{4-7}$$

$$y_t = Cx_t + Du_t + w_t. \tag{4-8}$$

We formulate the state transition equation as:

$$x_{t+1} = Ax_t + Bu_t + v_t = \tag{4-9}$$

$$\begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_{t+1} = \begin{bmatrix} g & 0 & 0 & 0 & 0 & 0 & & \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos(\omega\tau) & \frac{\sin(\omega\tau)}{\omega} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega\sin(\omega\tau) & \cos(\omega\tau) \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_t + \begin{bmatrix} \zeta \\ \nu_1 \\ \nu_2 \\ \nu_3 \\ \kappa \\ 0 \\ 0 \\ 0 \end{bmatrix}_t .$$

Here τ denotes the sampling time. The relation between the observations and the states of

the system can be described as:

$$y_t = Cx_t + w_t =$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t = \begin{bmatrix} a_1 & 1 & 0 & 0 & d_1 & \mu_1 & b_1 & 0 \\ a_2 & 0 & 1 & 0 & d_2 & \mu_2 & b_2 & 0 \\ a_3 & 0 & 0 & 1 & d_3 & \mu_3 & b_3 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} \eta_{1,t} + c_1\epsilon_t \\ \eta_{2,t} + c_2\epsilon_t \\ \eta_{3,t} + c_3\epsilon_t \end{bmatrix}. \tag{4-10}$$

The noise covariance matrices are defined as follows:

$$E\left[v_t v_t^\top\right] = Q = \begin{bmatrix} \sigma_\zeta^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\nu,1}^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\nu,2}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\nu,3}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\kappa^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4-11}$$

$$E\left[w_t w_t^\top\right] = R = \begin{bmatrix} \sigma_{\eta,1}^2 + c_1^2\sigma_\epsilon^2 & c_1c_2\sigma_\epsilon^2 & c_1c_3\sigma_\epsilon^2 \\ c_2c_1\sigma_\epsilon^2 & \sigma_{\eta,2}^2 + c_2^2\sigma_\epsilon^2 & c_2c_3\sigma_\epsilon^2 \\ c_3c_1\sigma_\epsilon^2 & c_3c_2\sigma_\epsilon^2 & \sigma_{\eta,3}^2 + c_3^2\sigma_\epsilon^2 \end{bmatrix}. \tag{4-12}$$

Here, σ_ζ^2, $\sigma_{\nu,s}^2$, σ_κ^2, $\sigma_{\eta,s}^2$, σ_ϵ^2 denote the variance of ζ_t, $\nu_{s,t}$, κ_t, $\eta_{s,t}$ and ϵ_t, respectively.

4-3-2 Models for track circuit faults

After a model for track circuit measurements has been derived, we continue with models describing faults. In some cases, multiple spatial influences and types of degradation behavior are linked to one fault mode. In those cases, we see each possible combination of spatial dependencies and degradation behavior as a separate mode of the system. This leads to the modes distinguished in Table 4-1. Since the characteristics described in [79] are not based on actual data, it might be that the described characteristics are inaccurate in practice. If this is the case, other models have to be used, but the same methodology can be applied.

We build discrete-time state space models for all described faults. Similar to the general model, linear time-invariant state space models can be used, as presented in Eq. (4-7). All faults are unrelated to rain sensor measurements. Therefore, in the fault models the terms Bu_t and Du_t disappear. This leads to the following model:

$$x_{t+1} = Ax_t + v_t \tag{4-13}$$

$$y_t = Cx_t + w_t. \tag{4-14}$$

In general, degradation behavior is incorporated in the state-transition matrix A. Spatial influences are related to the observation matrix C, this matrix describes what outputs are

Table 4-1: Modes distinguished in the multiple-model approach.

Mode (θ)	Problem	Spatial influence	Degradation behavior
0	Healthy state	-	-
3a	Insulated joint defect	D_1	L
3b		D_1	E
4	Conductive objects	D_1	A
5	Mechanical defect	D_1	E
6	Electrical disturbances	D_2	I
7a	Ballast degradation	D_1	L
7b		D_1	E
7c		D_2	L
7d		D_2	E

influenced by the fault states. In this section we use two states: the state $x_{1,t}$ represents the absolute fault intensity and $x_{2,t}$ represents the degradation rate (the derivative of the absolute fault intensity with respect to time).

The occurrence of a fault with linear degradation behavior (L) can be seen as a step in the degradation rate: the degradation rate suddenly changes from 0 (for healthy behavior) to a positive value. In [24], it is described that such step functions can be described using two models. One model is used to describe a step using a large amount of process noise v_t and another is used to describe linear degradation behavior where the degradation rate remains constant. The system may transition from the healthy model (θ_{healthy}) to the step model (θ_{step}) with probability π_{step} and then transition to the model where the degradation rate remains constant with probability 1 in the next time step. When the fault occurs, this is modeled using a system with a large amount of process noise on the degradation rate:

$$x_{2,t+1} = x_{2,t} + v_t \quad \text{for } t = t_{\text{step}}, \tag{4-15}$$

where $E[v_t v_t^\top] = Q$ is very large. Directly after this large step, the system is described using a model for linear degradation behavior without any process noise

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t \quad \text{for } t > t_{\text{step}}. \tag{4-16}$$

For exponential degradation behavior (E), we use a model of an unstable linear system:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & \tau \\ 0 & f_\theta \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t + v_t. \tag{4-17}$$

Here, f_θ is a parameter denoting the pole location of the exponential degradation behavior. The parameter f_θ is larger than 1 (since the system is unstable) and is different for each type of fault θ.

Abrupt faults (A) can be described using two models [24]. One model is used to describe a step using a large amount of process noise v_t and another is used to describe degradation behavior where the fault intensity remains constant. Similar to linear fault development, the system may transition from the healthy model (θ_{healthy}) to the step model (θ_{step}) with probability π_{step} and then transition to the model where the fault intensity remains constant with probability 1 in the next time step. When the fault occurs, this is modeled using a system with a large amount of process noise:

$$x_{1,t+1} = x_{1,t} + v_t \quad \text{for } t = t_{\text{step}}, \tag{4-18}$$

where $E[v_t v_t^\top] = Q$ is very large. Directly after this large step, the system is described using a model without any process noise

$$x_{1,t+1} = x_{1,t} \quad \text{for } t > t_{\text{step}}. \tag{4-19}$$

Intermittent faults (I) appear as spikes in the data. A common method of modeling such sudden outliers in a multiple-model approach is using a model with a large amount of measurement noise [24]:

$$y_t = w_t = \begin{bmatrix} w_{1,t} \\ w_{2,t} \\ w_{3,t} \end{bmatrix}. \tag{4-20}$$

where $E[w_{1,t} w_{1,t}^\top] = r_1$, $E[w_{2,t} w_{2,t}^\top] = r_2$ and $E[w_{1,t} w_{2,t}^\top] = E[w_{2,t} w_{1,t}^\top] = r_{1,2}$ are very large. If a spike occurs, the model with large amount of measurement noise is able to explain those spikes, while the healthy model cannot. In order to incorporate the intermittent property, transitions from an intermittent fault to the healthy state are allowed, as well as that the system remains in the fault mode (the mode with the large amount of measurement noise). In fact, the period between spikes can be described using the model of healthy behavior.

Now we have derived state transition equations for all types of degradation behavior, we relate the states to the observed outputs, by means of the C matrix. For faults that do affect only the track circuit to monitor, the following observation equation can be used:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t = Cx_t + w_t = \begin{bmatrix} 1 & 0 & \\ 0 & 0 & \dots \\ 0 & 0 & \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \end{bmatrix}_t + w_t. \tag{4-21}$$

For faults that are present at all sections on the same track, one could use

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t = Cx_t + w_t = \begin{bmatrix} 1 & 0 & \\ 1 & 0 & \dots \\ 0 & 0 & \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \end{bmatrix}_t + w_t. \tag{4-22}$$

Here the state representing the fault intensity $x_{1,t}$ is additive to all track circuits on the same track.

4-3-3 Transitions between modes

Next to specifying a linear state space model for each mode, we also need to specify the transitions that are allowed between models: the mode transition matrix. Except in the case

of intermittent faults, we only allow transitions from healthy behavior to faulty behavior and do not allow transitions from a fault mode to the healthy mode. The assumption of prohibiting transitions from a faulty mode to an healthy mode seems reasonable, since without repairs faults do not disappear. In practice, after a repair, condition monitoring can be reset. If we do assume that faults do not disappear by themselves, transitions between faults should be ruled out as well, since this would imply one fault that disappears while another appears.

We have visualized the transitions between the models in Figure 4-3. In total 13 models are used. One model describing healthy behavior, nine models describing a particular form of faulty behavior and three auxiliary models. These auxiliary models describe a sudden change in the measured current (related to a step in the fault intensity) or in the derivative of the current (related to a step in the degradation rate), as described in the previous section. In the figure, the allowed transitions are shown as well, which are assumed to have the Markov property. The transition probabilities from mode to mode can be calibrated using data from past cases.

4-4 Practical problems and solutions

Some practical problems are encountered when one wants to implement the presented approach in practice. In this section, we describe the problems and propose solutions.

First of all, the model for linear ballast degradation affecting only one section ($\theta = 7a$) and the model for insulated joint defects with linear degradation behavior ($\theta = 3a$) are equal. This makes it impossible to distinguish between these fault modes using the multiple-model approach. However, insulated joint defects have a shorter time from incipient of the fault to failure than ballast degradation has. It is possible to distinguish between the two faults using an additional classifier. We use a Bayes classifier[3] with only one feature: the derivative of the current with respect to time. The absolute value of the derivative of the current will be larger for an insulated joint defect than for linear ballast degradation, since the degradation rate of linear ballast degradation is much slower than that of insulated joint defects. The Bayes classifier assigns a probability to the model according to Bayes rule. Hereby, we assume that the derivative of the current comes from a normal distribution where the mean and variance for insulated joint defects is different from those of linear ballast degradation.

Second, the models derived in Section 4-3-1 and Section 4-3-2 are unobservable and some parameters are unidentifiable. Therefore these models cannot be used directly. We propose to use other realizations of the derived models that are observable, where all parameters are identifiable, and that obey the same system dynamics as the derived modes. We have derived the new realizations in Appendix A. The parameters in these models can be identified using grey-box identification. Some parameters are dependent on the actual case, such as the nominal currents of the outputs $\mu_{s,t}$. These parameters can be identified during the learning period of the system: a short time after installation of a new track circuit where the system is still healthy. Other parameters, such as parameters used to describe fault progression, can be estimated using data from previous cases where faults were present.

[3] Various classifiers could be used for this very simple classification task, but we are most familiar with the Bayesian classifier.

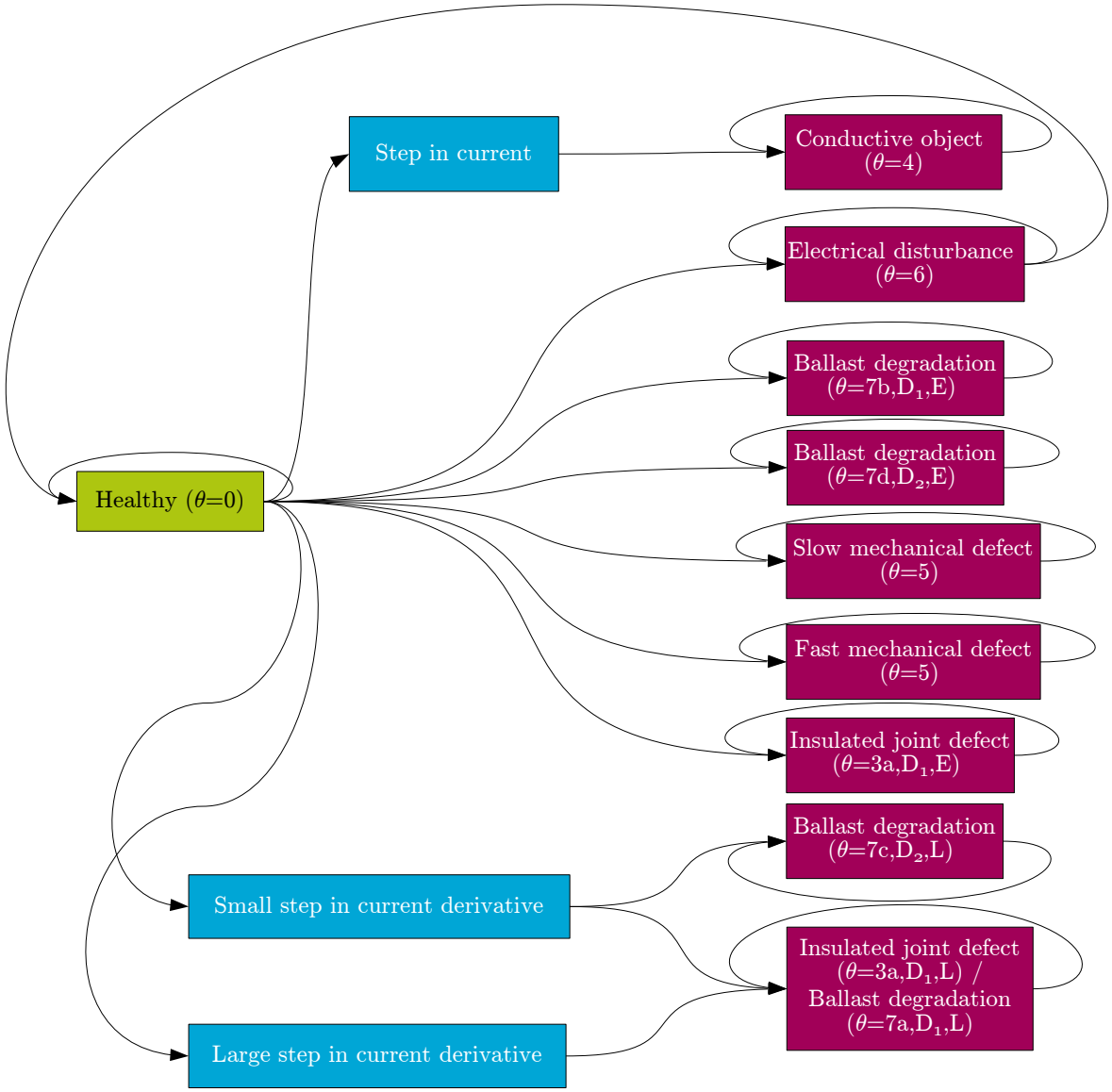


Figure 4-3: Models and allowed transitions used for track circuit condition monitoring.

Lastly, for healthy behavior, the system might be described using other models as well. In fact, the model of healthy behavior is nested in the model of, for example, linear ballast degradation. When the fault intensity $x_{1,t}$ and degradation rate $x_{2,t}$ are zero, this model reduces to the model for the healthy state. This has an intuitive explanation: healthy behavior can be seen as a fault with linear degradation behavior, where the fault intensity is zero and remains this for the entire sequence, i.e. the degradation rate (derivative of the fault intensity with respect to time) is zero as well. In fact, healthy behavior is just a special case of a system experiencing linear degradation behavior. In general, the system will stay in the healthy mode ($\theta = 0$) if there is no evidence in favor of another model, but sometimes the multiple-model filter assigns a higher likelihood to linear ballast degradation. This is caused by the fact that as time progresses, the effect of the fact that the system was initially healthy on the estimated current mode diminishes. To reduce the number of false alarms due to this effect, we can assign zero probability to faulty models when the current is constant or increasing. Only when the current is decreasing, we assume a fault is present. Instead of making a hard decision, we assign a probability to whether there is a fault present. We call this soft thresholding and we have illustrated the concept in Figure 4-4. We use a Gaussian cumulative density function to assign a probability to a fault. When the derivative of the current with respect to time is zero, the probability of a fault is very small and the probability of healthy behavior is almost 100 % and vice versa.

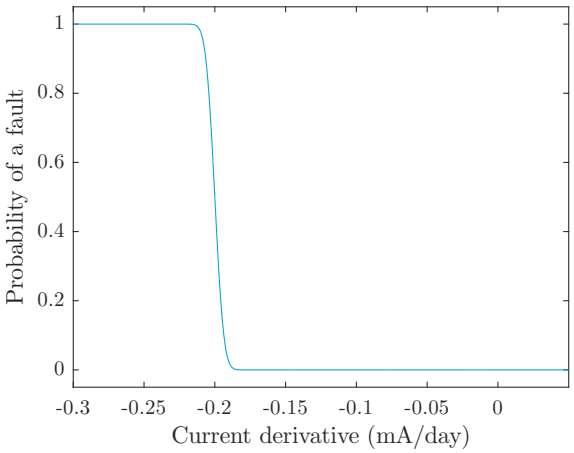


Figure 4-4: Probability assigned to a fault based on the estimated current derivative.

4-5 Conclusions

In this chapter, it was discussed how a multiple-model approach can be applied to the track circuit case. We have started by analyzing the problem and the signals we suggest to use. It was proposed to combine nearby track circuit measurements in order to let the system be described using the discrete health states of the monitored track circuit only. We have proposed to use the median since it is resistant to outliers. After having described the preprocessing stage, we have derived models for the track circuit case. The development of these models was split into the development of a model describing track circuit measurements in general and a model describing the faults. Next, we have determined the allowed transitions and decided to only allow transitions from healthy to faulty modes, except for intermittent faults. In addition, three practical problems were discussed and solutions were proposed. First of all, models for some fault modes are equal, which makes it impossible to distinguish between these fault modes using our approach. Therefore, we have proposed to use a Bayesian classifier to make the distinction based on the degradation rate. Second, the derived models are unobservable and we have indicated that this can be addressed by transforming the models to observable realizations. Lastly, we have described how the number of false alarms can be

reduced in cases where the estimated fault intensity is decreasing. A soft threshold was used for this purpose.

One small topic for future work arises from this chapter. In Section 4-1, we have described that the current flowing through the relays of the track circuits is sampled 50 times a day equidistantly. When the section is occupied at the time of sampling, we have suggested to use a measurement of the current just before the section became occupied. Another solution would be to not use the measurement for that time step for condition monitoring. It can be investigated whether this yields a better performance.

Chapter 5

Case study: a small railway network

In Chapter 4, we have shown how to apply a multiple-model approach to track circuit condition monitoring. In this chapter, we test this approach in a small case study. In Section 5-1, we describe the case we use. Next, we describe how we implement our proposed solution in Section 5-2. We evaluate the fault classification performance based on 1000 simulated cases in Section 5-3 and in Section 5-4 we compare the performance with that of a neural network that has been developed for the same problem. The predictive performance of our approach is evaluated in Section 5-5. In Section 5-6, we show the fault classification and prediction output for a typical case. Afterwards, we analyze what factors cause uncertainty in the remaining useful life prediction in Section 5-7. In Section 5-8, the main conclusions of this chapter are outlined.

5-1 Case description

In this case study, we consider a small railway network with track circuits to test our approach. For large networks, weather influences cannot be seen as homogeneous over the network, while for small networks they can. Testing the approach on a larger network would thus require accurate modeling of weather influences (including their spatial dependencies) to make the analysis sensible. Due to time constraints, we were not able to build a simulation where all spatial dependencies of weather influences were accurately incorporated. Therefore, we test the approach on a small network. The network we use is depicted in Figure 5-1. Our aim is to detect whether a fault is present at section 1. For this we have the following information:

- Measurements $I_{1,t}, I_{2,t}, I_{3,t}$ of the current flowing through the relays of the three track circuits.
- Measurements u_t of a rain sensor located in between the considered sections.
- The fact that no faults are present at track circuits 2 and 3 that do not influence section 1 as well. This means section 2 and 3 may suffer from electrical disturbances and ballast

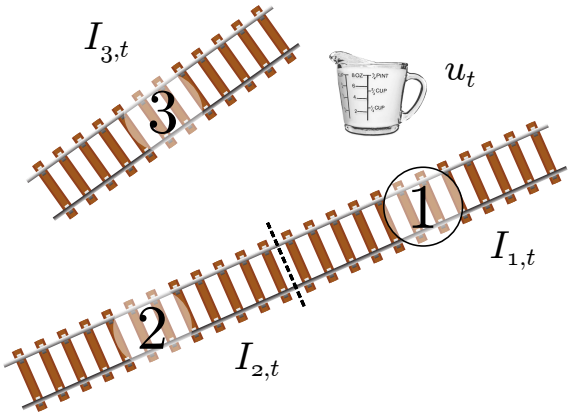


Figure 5-1: The railway network considered in the case study.

degradation, since these influence may influence track circuit 1 as well, but do not suffer from insulated joint defects, conductive objects, or mechanical defects.

- The fact that a track circuit can only suffer from one fault.

To generate data for the development and validation of our condition monitoring program, we use the simulation described in [19]. The simulation has been developed on a stand-alone basis and does not use the models we have developed for our multiple-model approach. The simulation is based on the observed characteristics of real track circuit measurements of three neighboring sections and the characteristics of faults described in [79]. In the simulation we simulate a period of two years. It is assumed that faults start between the 73th day (10 % of the simulated two years) and the 219th day (30 % of the two-year period) and the track circuits are fault-free before this point.

5-2 Implementation of our approach

We implement the proposed approach in MATLAB. For diagnostics, we make use of the interacting multiple-model filter contained in the extended Kalman filter/unscented Kalman filter toolbox [25]. We perform system identification on the system during the first 73 days after installation, where the system is (assumed to be) still healthy[1]. For prognostics for each case, we use 1000 runs[2] in our Monte Carlo approach to estimate the remaining useful life[3]. We use kernel density estimation to obtain estimates of the probability density function and the cumulative density function of the remaining useful life [47, 60]. For this purpose, we have used the MATLAB function `ksdensity(T)` where we used a Gaussian kernel and automated bandwidth selection. Based on the estimates of the probability density function, the 95 % confidence interval is obtained.

[1] We call this the learning period.

[2] The word 'run' is used to denote one prediction used in the Monte Carlo approach and not to denote the simulation of the data for a case in this case study.

[3] We have tested the effect of increasing the number of runs used in our Monte Carlo approach. This analysis can be found in Section 2 of Appendix B. It turns out that with 1000 runs a reasonable performance was obtained at a medium computational cost.

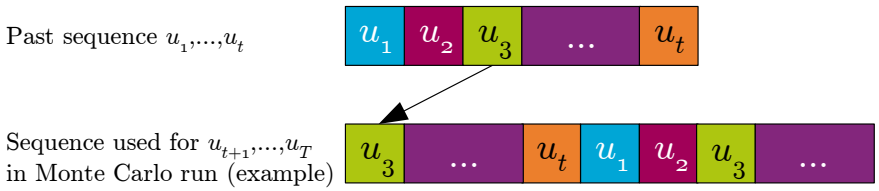


Figure 5-2: Circular shifted input sequence.

For each run in our Monte Carlo approach, we take into account the uncertainty in the estimates of the current state, uncertainty in the diagnostic classification, uncertain future inputs, uncertainty in the model parameters and uncertainty in future process and measurement noise. In each run, we draw these quantities randomly from the distributions describing these uncertainties. The model is drawn randomly from the discrete distribution of models used for diagnostic classification, where each model gets a weight equal to the probability of being active according to the implemented diagnostic approach. This means that when the interacting multiple-model filter assigns a 90 % probability to ballast degradation and a 10 % probability to mechanical defects, we pick the ballast degradation model roughly nine out of the ten runs and the mechanical defect model only once in ten runs. The interacting multiple-model filter estimates, next to the active model, also the continuous states of all models and their covariance. For each run, we draw the states used for each run randomly from a Gaussian with as mean the estimated states $\bar{x}_t$ and as covariance the covariance matrix P_t according to the Kalman filter for that specific model. Input uncertainty from the weather is handled by drawing a randomly circular shifted version of past weather measurements $u_1, ..., u_t$ and using this new sequence as future inputs $u_{t+1}, ..., u_T$. We have illustrated this shifting process in Figure 5-2. We repeat the shifted sequences up to time T. The number of time steps the input sequence is shifted is based on a random integer between 1 and t (inclusive). We use circular shifting to maintain the serial correlation present in the input sequence. Parameter uncertainty is taken into account by drawing identified parameters randomly from a normal distribution with the mean and variance according to the system identification estimates for each run. Lastly, input and process noise is drawn based on the process and measurement covariance matrices Q' and R of the model used in this run. We use the measurement covariance matrix R that is identified during the learning phase. The process noise covariance matrix Q' is proportional to the process noise covariance matrix Q identified during the learning phase, i.e. $Q' = \delta Q$. The scalar δ is calibrated using data from previous cases. Using δ, it is possible to account for small modeling errors. When adequately calibrated, this will improve the accuracy of the uncertainty approximation. Correcting the process noise matrix is common practice in condition monitoring, see e.g. [44, 69].

5-3 Fault classification performance

We test our proposed solution on 1000 simulated cases. For each case, a fault is selected and two years of data is simulated. Since we distinguish 6 health states (healthy behavior and 5 faults), this leads to 166 or 167 simulated cases per health state. Our implemented condition monitoring approach has to identify the health state of the system. When the fault intensity

Table 5-1: Fault classification performance based on 1000 simulated cases in total.

Fault type	Mean logarithmic loss likelihood	Accuracy
Healthy	0.01	99.8 %
Insulated joint defect	0.00	100.0 %
Conductive object	0.04	99.4 %
Mechanical defect	0.27	98.2 %
Electrical disturbance	0.06	99.1 %
Ballast degradation	0.10	97.8 %
Overall	**0.08**	**99.1 %**

in our simulation is between 15 % and 100 % (failure), we evaluate the classification of the fault according to two criteria[4]. As a first criterion, we use the logarithmic loss likelihood. The logarithmic loss likelihood is formulated as follows:

$$\mathcal{L} = -\log p\left(\theta_t\right), \tag{5-1}$$

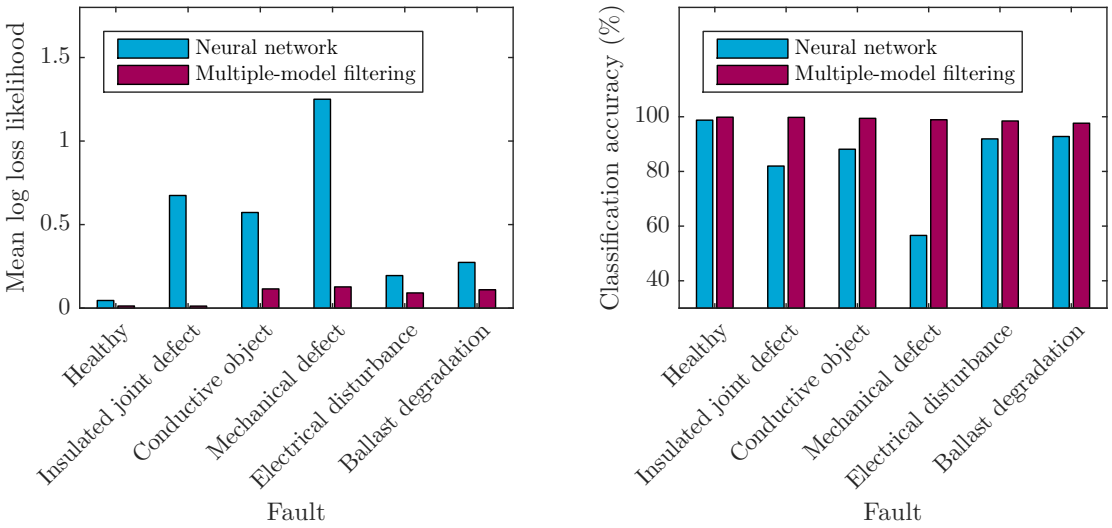
where $p\left(\theta_t\right)$ is the probability assigned to the correct mode θ_t. Furthermore, we obtain hard classification labels by selecting the most likely mode and calculate the classification accuracy, which is the fraction of correctly classified observations. Where the classification accuracy metric does not take the confidence of the prediction into account, the logarithmic loss likelihood does and 'punishes' overconfident predictions relatively severe. Therefore this combination of metrics supports the evaluation of the overall performance. Both metrics are used in [19] as well, which facilitates the comparison of our results with those found in [19].

In Table 5-1, the mean logarithmic loss likelihood and the classification accuracy are presented. From the table we can conclude that, in general, the classification performance is very good: in 99.1 % of the cases the system correctly diagnoses the track circuit. The implemented condition monitoring approach achieves the lowest logarithmic loss likelihood, and therefore the best performance, for classifying healthy sequences. The approach has most difficulty with detecting mechanical defects: the worst logarithmic loss likelihood and relative accuracy is obtained for this class of faults. The reason that the logarithmic loss likelihood is very high for this class is that for one in the 167 cases where the track circuit suffered from a mechanical defect, almost zero probability was assigned to the correct mechanical defect class. In fact the approach was very confident about its classification, but was wrong. This case has a mean logarithmic loss likelihood of 30, which has a large influence on the presented statistics. To reduce the effect of mistakes due to overconfidence, one could implement a lower bound on the model probability of each model. This would maximize the amount of likelihood that can be lost.

5-4 Fault classification performance compared with that of a neural network

In order to put the performance of our diagnostic approach in perspective, we compare it with an approach where neural networks are used. This diagnostic approach is described in

[4]For healthy sequences we calculate these two criteria over the entire simulation time of two years.



(a) Mean logarithmic loss likelihood.

(b) Classification accuracy.

Figure 5-3: Fault classification performance compared with a neural network.

[19] and makes use of the long short-term neural network architecture. The neural network is trained using 2700 cases. We use the same criteria for evaluation as in the previous section: the logarithmic loss likelihood and classification accuracy over the area where the fault intensity is above 15 % and the fault has not yet caused a failure. The neural network operates at a sampling frequency of 25 samples per day, while the implemented multiple-model approach operates using 50 samples per day[5]. Therefore, we use subsampling and decimate the simulation output with a factor 2 before they are fed into the neural network. In fact this is preprocessing. Since the neural network diagnoses the track circuit 25 times a day, while the multiple-model filter outputs a diagnostic classification 50 times a day, we perform subsampling of the classification output of the multiple-model filter (decimation with a factor 2). In fact this postprocessing. This allows us to compare the classification output of both methods with a frequency of 25 samples a day. A graphical representation of this set-up can be found in Appendix C. We test the fault classification performance on a validation data set with 300 cases (50 cases for each health state).

The results are shown in Figure 5-3. From the figure, we can conclude that our multiple-model filtering approach outperforms the neural network developed for the same purpose. In terms of fault classification accuracy, the difference is the largest for mechanical defects and smallest for the detection of healthy behavior. For the mean logarithmic loss likelihood, the same picture emerges. We have not compared the computational performance of both methods, since both methods are implemented on a different platform: the multiple-model approach runs on MATLAB, while the neural network is implemented using Torch, which is a specialized library for neural networks with an underlying C implementation.

From the comparison we may conclude that in this particular case a multiple-model approach outperformed that of a neural network. It is important to note that this may not only be

[5] We have also tested a neural network operating at 50 samples a day, but it turned out that the performance was much worse than a neural network operating at 25 samples a day.

caused by the method itself, but that the actual implementation of both methods may explain a part of this difference. For example, the number of neurons used and the form of the activation functions in the neural network might be suboptimal. Furthermore, it would be possible to train the neural network with more data. This might lead to an improved performance of the neural network. In contrast, the effect of using more data on the performance of the multiple-model approach will be much smaller, since only a small part of the models used is estimated from data. The main difference between the neural network approach and the multiple-model approach is that the first does not use prior knowledge, while the latter does. If new prior knowledge becomes available, this might be used to improve the multiple-model approach, while it will not effect the performance of the neural network. The presented results suggest that the incorporation of prior knowledge might be beneficial for track circuit condition monitoring.

5-5 Prognostic performance

When our condition monitoring approach has detected a fault, predictions for the remaining useful life can be made. In this case study, we only make predictions for faults that evolve gradually over time, i.e. insulated joint defects, mechanical defects and ballast degradation. Our prognostic approach is able to make a prediction every time step[6], but to limit the computation time, we evaluate predictions only each time the current level of the monitored track circuit has decreased with[7] 0.03 A. In practice, this means roughly 6 predictions are made for each simulated case. In total, 3165 predictions are used for performance evaluation.

We evaluate the prognostic performance using the point prediction accuracy, relative accuracy and the convergence of the 95 % confidence interval [64]. First of all, we use the point prediction accuracy, which is the difference between the predicted remaining useful life $\hat{T}$ and the actual remaining useful life T. Next to the absolute prediction error $\left|T - \hat{T}\right|$, we evaluate the relative accuracy of the point prediction:

$$\mathcal{A} = \frac{\left|T - \hat{T}\right|}{T}. \tag{5-2}$$

We make a prediction of the remaining useful life by calculating the median of the estimated remaining useful life distribution. We choose the median, since the performance of the point predictions is evaluated using an absolute loss function. It can be shown that under absolute loss, the median of the predicted distribution is the optimal estimator [23].

As has been pointed out in [64], convergence of the predictions is important for prognostics. To test whether the predictions converge over time, we inspect the width of the 95 % confidence interval for the first prediction and the last prediction made before failure. The latter should be much smaller than the first.

The main performance characteristics for the predictions are shown in Table 5-2. From Table 5-2a we conclude that the absolute error of the point predictions averaged over all

[6] The time required for a prediction is about 10 seconds. This is much shorter than the sampling interval, which is about 29 minutes.

[7] This offers a fairer evaluation than using a fixed amount of time between predictions, since some faults evolve much faster over time.

Table 5-2: Prognostic performance based on 3156 predictions for the remaining useful life in total.

(a) Absolute prediction error.

Fault type	Absolute prediction error (days)	Absolute prediction error first prediction (days)	Absolute prediction error last prediction (days)
Insulated joint defect	3.22	10.73	1.19
Mechanical defect	1.70	9.80	0.15
Ballast degradation	15.92	43.97	6.81
Overall	**6.92**	**21.43**	**2.71**

(b) Prediction accuracy and width 95 % confidence interval.

Fault type	Prediction accuracy	Width 95 % confidence interval first prediction (days)	Width 95 % confidence interval last prediction (days)
Insulated joint defect	82 %	63.59	5.06
Mechanical defect	81 %	66.17	0.46
Ballast degradation	80 %	184.27	46.39
Overall	**81 %**	**104.43**	**17.22**

predictions is 6.92 days. Furthermore, we see that the prediction error decreases over time: when getting closer to the moment of actual failure, the prediction error decreases. In Table 5-2b it can be seen that the predictions have an average accuracy of 81 %. Furthermore, the confidence interval shrinks over time: the first prediction for each case has an average uncertainty of more than three months, but just before failure, the confidence interval shrinks to only two weeks. In general, we see that predictions for ballast degradation are less accurate than predictions for other fault modes. Probably this is due to the fact that ballast degradation occurs over a long period of time (roughly a year), while insulated joint defects and mechanical defects evolve much faster.

We have tested the validity of the 95 % confidence interval of the predicted remaining useful life. The actual remaining useful life falls within the confidence interval in 94.89 % of the cases. This is close enough to the nominal value of 95 % for all practical purposes. We have also tested the Monte Carlo method with the identified process noise covariance matrix Q, instead of its scaled variant $Q' = \delta Q$, i.e. $\delta = 1$. When using this process noise covariance matrix, in 93.9 % of the cases the actual remaining useful life falls within the 95 % confidence interval of the prediction. The predictions and prognostic performance using Q are almost identical to the results obtained using Q'. This means that the effect of the prediction correction is small in this case study. The prognostic results obtained using Q can be found in Section 3 of Appendix B.

5-6 Fault classification and prediction output for a typical case

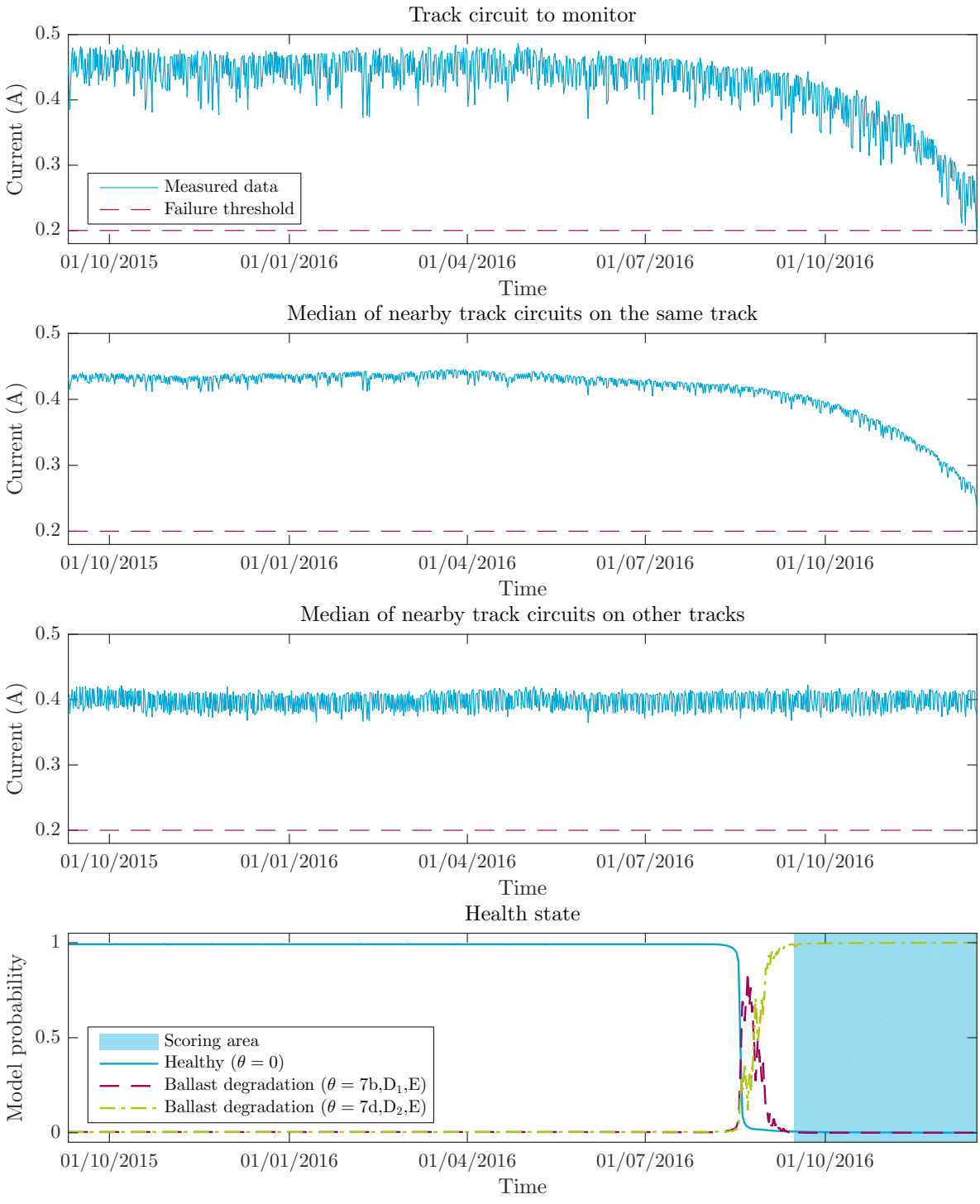
In this section, we illustrate the results of the case study by inspecting the condition monitoring output for one of the 1000 simulated cases. In this example case, ballast degradation is present. The example is representative for the rest of the cases.

We start with diagnostics, where the probability that a certain fault mode is active is assigned at each time step. The diagnostic output for the example case is shown in Figure 5-4. In the upper three graphs, the measured current is shown (solid blue line) and the failure threshold is drawn (dashed purple line). In the bottom graph, the output of the diagnostic system is shown. At each time step, the model probability is plotted. Models with a very low probability assigned ($p(\theta_t) < 0.02$ for all t) are omitted for clarity. From the figure, we can see that the program first assigns the highest probability to healthy system behavior. Later, when the current level of the monitored track circuit start to decrease, the probability of healthy behavior decreases and the probability of ballast degradation increases. During the time the fault intensity is above 15 %, the assigned probability to this fault mode remains slightly less than 100 %. This is the desired behavior.

When ballast degradation is detected, predictions are made for the remaining useful life. An example of such a prediction for our example case is shown in Figure 5-5. In the top graph the measured current of the track circuit to monitor and the predictions of the future current are shown. In the middle plot, the probability density function is given of the time to failure. In the lower plot, the cumulative density function of the remaining useful life is shown. From the figure, we can see that the prediction is quite uncertain, but that the 95 % confidence interval includes the moment of actual failure. Furthermore, we see that the distribution of the estimated remaining useful life is highly non-symmetric. This supports the use of the Monte-Carlo approach: using the Monte-Carlo approach it is possible to obtain such non-symmetric distributions.

In Figure 5-6a we see an example of the predictions for the remaining useful life over time. The solid blue line represents the actual remaining useful life (at time of failure, the remaining useful life is zero). The solid purple marks with corresponding 95 % confidence interval represent the predictions that are made for the remaining useful life at each time-instant. From the figure, we can see that in general, the actual remaining useful life is included in the 95 % confidence interval and that the predictions become more accurate over time (the confidence interval shrinks).

In Figure 5-6b, the absolute error of the median prediction for our example case is shown. We see that the absolute prediction error decreases over time. Only at the end there exists a small increase in the absolute prediction error. In Figure 5-6c, the evolution of the width of the 95 % confidence interval over time is shown. From the dashed blue line, we conclude that the width of the confidence interval decreases over time, as we expect. In Figure 5-6d, the likelihood of failure at T over time is shown. The likelihood is the height of the probability density interval evaluated at the actual time of failure T. This likelihood is a measure of accuracy as well: when the confidence interval is wide the probability density function is low everywhere; when the confidence bands are small, the likelihood is high within that region, but very low outside it. We see that the likelihood (shown with the dashed blue line) increases over time. From the plots, we conclude that the predictions converge for this example case.



Unlikely models are not plotted for clarity

Figure 5-4: Soft classification over time for the example case.

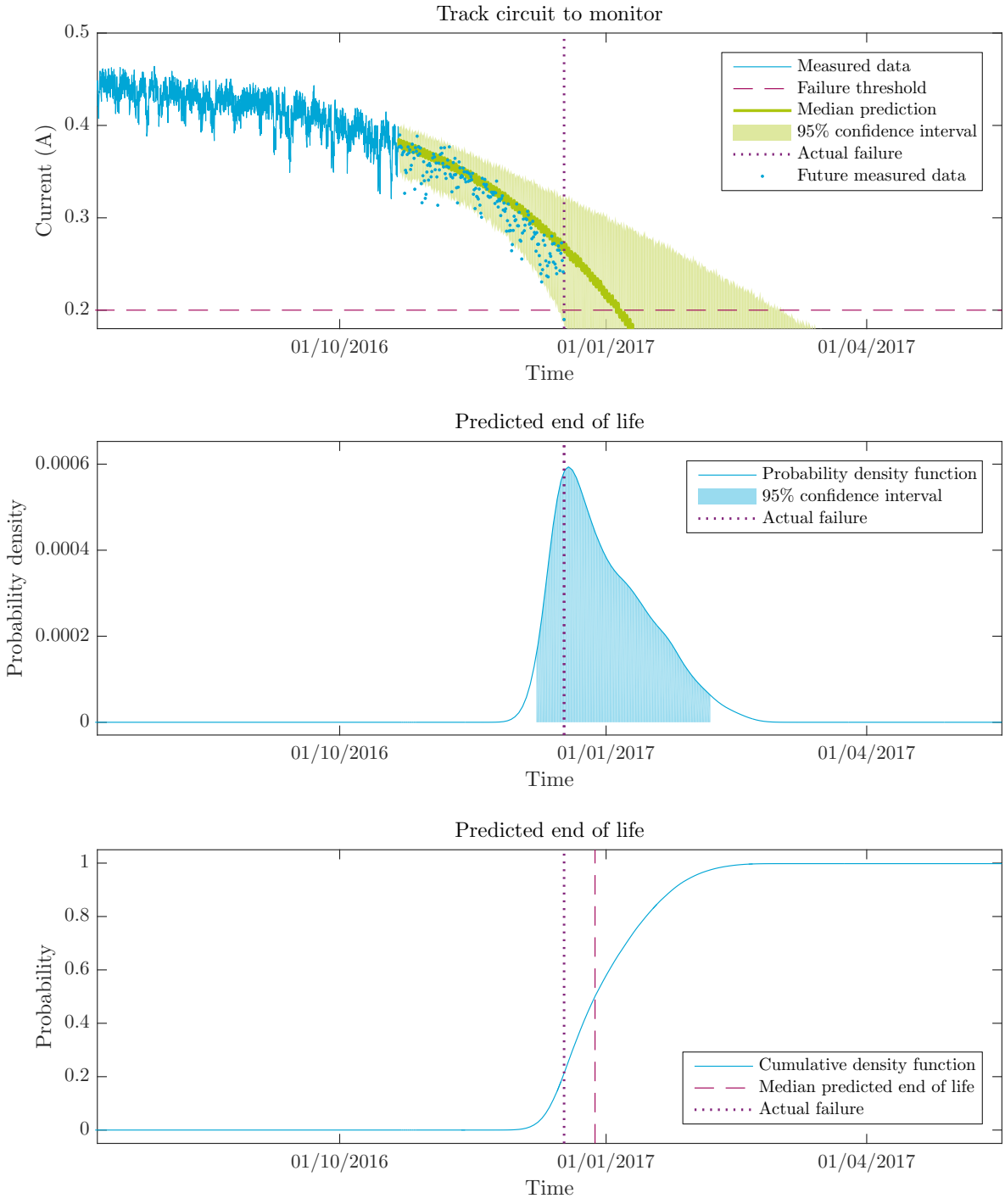
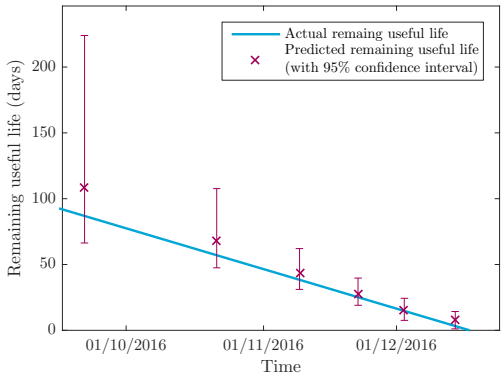
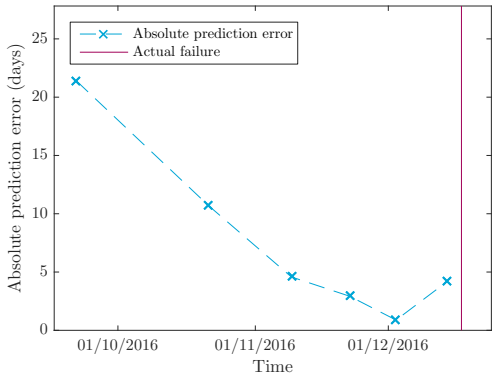


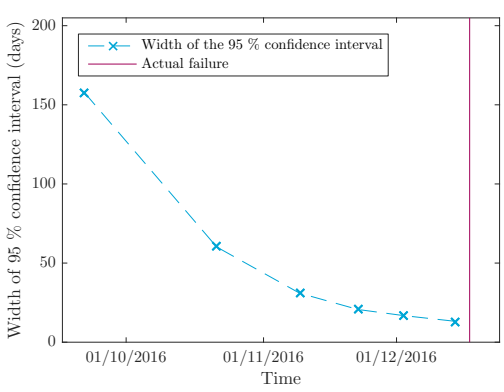
Figure 5-5: Remaining useful life prediction for the example case.



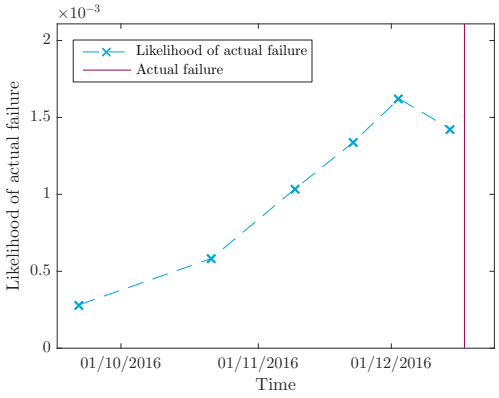
(a) Predictions over time.



(b) Absolute error over time.

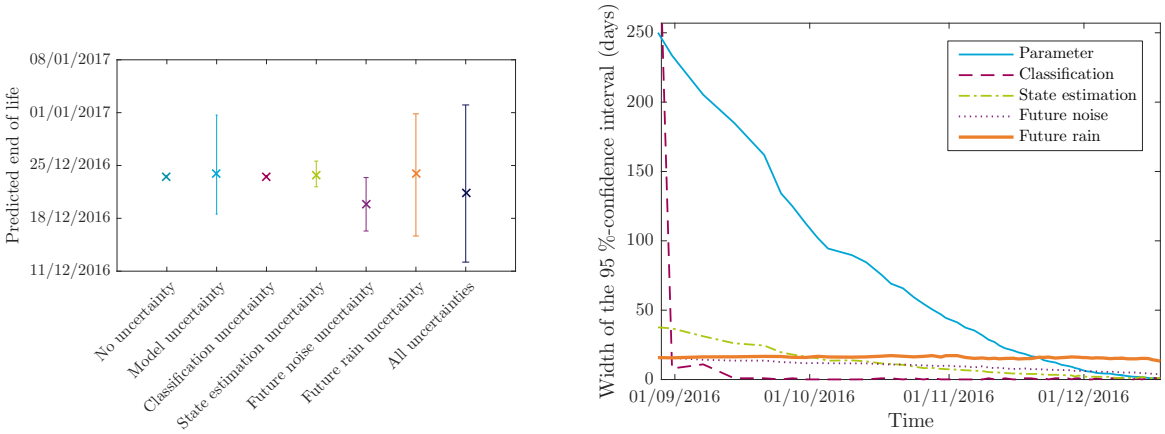


(c) Width of the 95 % confidence interval over time.



(d) Likelihood of actual failure.

Figure 5-6: Prediction characteristics over time for the example case.



(a) Effect of uncertainty sources on the size of the 95% confidence interval for a single prediction.

(b) Effect of uncertainty sources on the size of the 95% confidence interval over time.

Figure 5-7: Effect of uncertainty sources on the 95% confidence interval of the remaining useful life prediction for the example case.

5-7 Factors causing prediction uncertainty

In Figure 5-5, a typical prediction of the remaining useful life and the corresponding uncertainty is shown. In this section, we are interested in what factors cause the uncertainty in the prediction. To analyze the factors, we study the effect of each uncertainty source on the 95 % confidence interval of the predicted remaining useful life. We distinguish five sources contributing to uncertainty: uncertainty in the estimated parameters in the models, uncertainty caused by doubt about the correct fault mode, state estimation uncertainty, uncertainty related to future process and measurement noise, i.e. the realizations of $w_{t+1}, ..., w_T$ and $v_{t+1}, ...v_T$, and uncertainty about future inputs $u_{t+1}, ..., u_T$ (in our case future rain). We set all except one uncertainty sources to zero and generate predictions using the only non-zero source of uncertainty. We perform this analysis for all described uncertainty sources. In Figure 5-7a the effect of the factors on the 95 % confidence interval of the remaining useful life prediction for the example case are shown. From the plot, we conclude that in this example case the influence of classification uncertainty is relatively small; we see that in most cases. Probably, this is due to the fact that our diagnostic approach is often quite confident about its classification: if a probability of more than 97.5 % is assigned to one particular fault mode, it is likely that the 95 % confidence interval is not significantly affected, since predictions for other fault modes belong to the upper or lower 2.5 % of the estimated remaining useful life distribution.

It is interesting to see that some uncertainties have a significant effect on the point prediction. For example, future noise uncertainty leads to a point prediction where the remaining useful life is shorter than without taking future uncertainty into account. This is in line with our motivation in Section 3-5, where we stated that uncertainty should be taken into account since it might influence not only the uncertainty around the prediction, but the point prediction as well. This effect can be explained as follows. The track circuit fails the first time the current reaches the failure threshold. When the current behaves erratically due to noise, more spikes exist which might exceed this failure threshold leading to an early failure. We

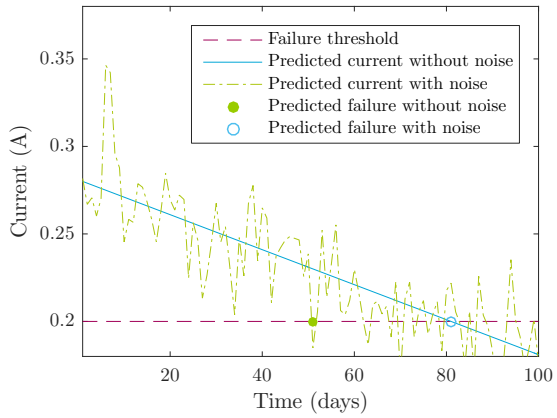


Figure 5-8: Illustration of the effect of incorporating future noise on the predicted remaining useful life.

have illustrated this in Figure 5-8, where we show a prediction for the current of a track circuit with and without taking future noise into account. From the figure, it is clear that the signal where future noise is taken into account crosses the failure threshold multiple times before the signal without noise crosses the failure threshold. This shows that, in general, taking noise uncertainty into account leads to a shorter predicted remaining useful life, which is in line with the results found.

We have also investigated how the factors evolve over time. A typical plot of how the width of the 95 % confidence interval evolves over time for each uncertainty source is shown in Figure 5-7b. Again, this plot is made for the example case, but representative for the rest of the cases. From the plot, we conclude that in order to improve predictions made a long time before failure, we have to reduce the amount of classification uncertainty and parameter uncertainty, since these factors have the largest influence on predictions more than a month before failure. In the figure, we see that the classification uncertainty suddenly decreases. This is due to the fact that when a probability of more than 97.5 % is assigned to one fault mode, other fault modes do often not effect the 95 % confidence interval, since they belong to the lower or upper 2.5 % of the probability density function of the remaining useful life. When the probability assigned to a fault mode is below 97.5 %, the 95 % confidence interval is affected by other fault modes as well, since these predictions fall inside the middle 95 %. The moment the probability assigned to a signal fault crosses the 97.5 % threshold, this causes an abrupt decrease in the classification uncertainty, which is visible in the figure. In this example case, the large amount of classification uncertainty around early predictions, is caused by diagnostics, which assigned a significant probability to healthy behavior. Since the remaining useful life of healthy behavior is unknown, the resulting classification uncertainty is large.

The absolute influence of weather uncertainty on the predicted remaining useful life is constant over time, but the relative importance increases over time. If we want to improve the prediction quality for predictions a few days before the end of the track circuit life, we have to incorporate information on weather forecasts into our Monte Carlo predictions.

This analysis reveals an important advantage of the multiple-model approach: the quan-

tification of factors underlying the prediction accuracy. Using these results, we know what factors lead to uncertainty in the predicted remaining useful life and how we can improve these predictions.

5-8 Conclusions

In this chapter we have shown how the proposed multiple-model approach performed in a simulation-based case study. The multiple-model approach is able to classify 99.1 % of the faults correctly. Furthermore, the fault classification performance of the multiple-model approach is better than that of a neural network developed for the same purpose. This suggests that the incorporation of prior knowledge might be beneficial for track circuit condition monitoring. Moreover, the multiple-model approach is able to make predictions for the remaining useful life with an average relative accuracy of 81 % and the uncertainty around those predictions decreases when the remaining useful life approaches zero. Lastly, we have analyzed what uncertainty sources had the largest influence on the prediction uncertainty. Using this analysis, we have found out how we can improve our predictions. It turned out that for prediction weeks or months before failure, parameter uncertainty and classification uncertainty have the largest influence. For predictions a few days before failure, uncertainty about future weather is the most important source. Furthermore, using the prediction uncertainty analysis we have shown that taking into account uncertainty has an influence on point predictions of the remaining useful life. This supports the use of a Monte Carlo approach.

The results in this chapter lead to two small topics for future work. First of all, we have seen in Section 5-3 that a single case with an overconfident prediction has a large on the mean logarithmic loss likelihood. Therefore, we suggest to introduce a lower bound on all model probabilities. Second, from Section 5-7 it followed that the largest uncertainty source for predictions a few days before failure is future weather uncertainty. We suggest to incorporate information on weather forecasts into our approach and to test whether test whether this indeed improves our predictions a few days ahead.

Chapter 6

Conclusions and future work

In this chapter, we critically reflect on the results found in this thesis. We start by presenting the main findings in Section 6-1. Next, we elaborate on the limitations of our research in Section 6-2. Based on these limitations, we make suggestions for future work in Section 6-3.

6-1 Main findings

In this thesis, we have presented a condition monitoring approach for track circuits based on the multiple-model methodology. Main advantages of the multiple-model approach are: the incorporation of prior knowledge, the fact that diagnostics and prognostics are integrated, and the capability of modeling sudden changes and shocks. We have shown that track circuits, including their faults, can be modeled as Markov jump linear systems. Using the efficient state estimation algorithms for Markov jump linear systems, the health state of the system can be determined and faults can be diagnosed. To make predictions for the remaining useful life, a Monte Carlo approach is proposed. By making multiple predictions using different initial conditions and noise realizations, the uncertainty in predictions for the remaining useful life can be quantified, which is useful information for decision making.

When the proposed approach was tested in a simulation-based case study, it turned out that the performance was remarkable. The implemented approach correctly diagnosed the system in 99.1 % of the cases and outperformed a neural network developed for the same purpose. Furthermore, the approach was able to generate predictions for the remaining useful life with a relative accuracy of 81 %. We have shown that our approach was able to quantify the uncertainty in the prediction of the remaining useful life. In addition, the method was able to quantify the effect of uncertainty sources on the remaining useful life prediction uncertainty. It turned out that for predictions months before the moment of functional system failure, parameter uncertainty and classification uncertainty are the most important factors. For predictions less than a week before failure, uncertainty about the weather is the most important source. We conclude that a multiple-model approach, combined with Monte Carlo-based predictions, is well suited to the track circuit condition monitoring problem.

In the future, this thesis together with [19] and [79] can be used to select a technique for track circuit condition monitoring. However, it will require a considerable amount of work before condition monitoring for track circuits is implemented, since the results of this thesis and [19, 79] have to be tested on real data. We believe that this thesis can be seen as one of the first steps towards achieving considerable less emergency repairs and breakdowns in 2030.

6-2 Limitations

There are some limitations to our research. Most of these limitations stem from the limited amount of data available.

First of all, we did not test the approach on a large railway network with track circuits. Although our framework is derived for large large networks with track circuits, due to time restrictions, in our case study we have used a smaller network. Testing the proposed approach on a larger network requires the simulation to be extended with realistic weather influences where spatial and temporal relations (especially of rain fall) are incorporated.

Second, in this thesis we have assumed that a track circuit can only suffer from one fault at a time. In practice, it might happen that two faults are present at the same time.

Third, we had to rely on prior knowledge of unknown reliability. The presented prior knowledge might be wrong or too simplistic for the real situation. Since no formal guarantees are derived in this thesis, it is not completely clear what the effect is of incorrect prior knowledge on the condition monitoring performance. It is also possible that new prior knowledge becomes available. This can be incorporated in a similar fashion as the prior knowledge used in this thesis. However, we do not know what the exact effect is of other prior knowledge on the condition monitoring performance. This means simulations and tests have to be performed using the newly developed system.

Lastly, we did not have measurements of track circuits suffering from a fault available. We only had measurements of three healthy track circuits. Therefore, we had to test our approach using a simulation-based case study. The general problem with simulation-based case studies is that only known characteristics are incorporated in the simulation and unknown phenomena or not. A simulation is always a simplification of reality and unmodeled phenomena might have a significant impact on the condition monitoring performance in practice. Therefore, it remains unknown how the proposed approach performs in reality.

6-3 Future work

In the previous section, we have pointed out some limitations of this study. In this section, we propose topics for future research, partly inspired by these limitations. First of all, we do suggestions for improvements and additional research based on the approach proposed in this thesis. These suggestions could be implemented in the short term (approximately three months). Second, we do suggestions for research topics related to our study that can be exercised in the mid term (approximately one year). Third, we suggest research directions for condition monitoring of track circuits in general, related to the long term. Lastly, we look at

a broader perspective and think about the applicability of the results presented in this thesis to other fields.

On the short term, we think our multiple-model approach could be improved using some minor modifications. We used the interacting multiple-model algorithm because of its excellent trade-off between computational complexity and state estimation performance. However, we are able to filter two years of data in 2 minutes per monitored track circuit, which is much faster than real-time. Therefore, choosing an algorithm that requires more computation time, but achieves a better performance might be a better choice. For example, filtering can be performed using the generalized pseudo Bayes 2, generalized pseudo Bayes 3, or generalized pseudo Bayes 4 algorithm [1, 11]. Furthermore, a numerically robust algorithm could be used. The implemented algorithm we used in our case study was based on the extended Kalman filter/unscented Kalman filter toolbox for MATLAB [25] and suffered from some numerical problems. The solutions proposed in [59] are a good starting point if one wants to develop such an algorithm themselves.

Next to these small topics for further research, substantial more research is required to test the applicability of the proposed approach to larger networks. Such a study requires a realistic simulation where the spatial properties of environmental influences are accurately modeled. Furthermore, this could also shed a light on the optimal number of nearby track circuits to use for condition monitoring. Furthermore, it is important to test the proposed approach on real data, instead of on simulated data. Unmodeled effects in the simulation or incorrect prior knowledge might have a large effect on the condition monitoring performance.

In the long term, we think research should be directed towards the influence of multiple faults on a single track circuit. We suggest performing field experiments for this purpose, since we expect that the amount of data available of track circuits suffering from multiple faults is much smaller than the amount of data available for track circuits suffering from a single fault. When information on the spatial influences and degradation behavior of these fault combinations is available, our multiple-model approach can be extended with models of fault combinations. Furthermore, we think a better condition monitoring solution can be achieved when multiple solutions are combined. Although our approach outperformed a neural network, combining the two might lead to a higher classification accuracy than that is achieved by the multiple-model approach alone. In practice, often combinations of classifiers outperform the single best classifier [29]. Therefore, we propose to combine the classification output of our multiple-model approach and the output of the neural network. Another suggestion might be to combine different estimation algorithms. In Chapter 3, we have concluded that the filtering performance of the interacting multiple models algorithm and the linear minimum mean square error estimator is comparable and that pruning strategies can be used as well. Combining the algorithms might increase the fault classification accuracy further. In turn, this might lead to better predictions, since the predictions for the remaining useful life are based on the outcomes of the fault classification stage.

The results of this thesis are not only applicable to the track circuit case, but can be applied to other areas as well. The application of track circuits was characterized by spatial dependencies. In this thesis we have proposed an approach whereby these dependencies are incorporated, while condition monitoring is performed for each track circuit separately, leading to a limited number of modes in the multiple-model approach. This technique might be applicable to other applications where spatial dependencies are important. One could think

of, for example, diagnosing water or gas networks. Next to applications in an engineering context, one could also think of economic applications. Markov switching models are often used for business cycle analysis (dating and predicting recessions and expansions), but often the business cycle is analyzed for one country or one continent without taking relations with others into account [32, 41]. Since the economies of countries get more and more entangled due to globalization, business cycle analysis might be improved using the techniques presented in this thesis.

Appendix A

Observable model realizations

In Section 4-3 we derived linear discrete-time state space systems for all modes of the system. As pointed out in Section 4-4, the described systems are not observable and not all parameters can identified. Therefore we have to transform our models to observable realizations. In this appendix we highlight how we form these observable realizations where all parameters are identifiable.

In Section A-1, we describe how to make the general model for track circuit measurements observable and all parameters identifiable. In Section A-2, we describe how we add the fault models to this model in order to create observable fault models.

A-1 An observable realization of the general model

We have the following state transition equations as defined in Section 4-3-1:

$$x_{t+1} = Ax_t + Bu_t + v_t = \tag{A-1}$$

$$\begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_{t+1} = \begin{bmatrix} g & 0 & 0 & 0 & 0 & 0 & & \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos(\omega\tau) & \frac{\sin(\omega\tau)}{\omega} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega\sin(\omega\tau) & \cos(\omega\tau) \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_t + \begin{bmatrix} \zeta \\ \nu_1 \\ \nu_2 \\ \nu_3 \\ \kappa \\ 0 \\ 0 \\ 0 \end{bmatrix}_t .$$

With the following observation equations:

$$y_t = Cx_t + w_t =$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t = \begin{bmatrix} a_1 & 1 & 0 & 0 & d_1 & \mu_1 & b_1 & 0 \\ a_2 & 0 & 1 & 0 & d_2 & \mu_2 & b_2 & 0 \\ a_3 & 0 & 0 & 1 & d_3 & \mu_3 & b_3 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \xi \\ 1 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} \eta_{1,t} + c_1\epsilon_t \\ \eta_{2,t} + c_2\epsilon_t \\ \eta_{3,t} + c_3\epsilon_t \end{bmatrix}. \tag{A-2}$$

The noise covariance matrices are defined as follows:

$$E\left[v_t v_t^\top\right] = \begin{bmatrix} \sigma_\zeta^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\nu,1}^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\nu,2}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\nu,3}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\kappa^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A-3}$$

$$E\left[w_t w_t^\top\right] = \begin{bmatrix} \sigma_{\eta,1}^2 + c_1^2\sigma_\epsilon^2 & c_1c_2\sigma_\epsilon^2 & c_1c_3\sigma_\epsilon^2 \\ c_2c_1\sigma_\epsilon^2 & \sigma_{\eta,2}^2 + c_2^2\sigma_\epsilon^2 & c_2c_3\sigma_\epsilon^2 \\ c_3c_1\sigma_\epsilon^2 & c_3c_2\sigma_\epsilon^2 & \sigma_{\eta,3}^2 + c_3^2\sigma_\epsilon^2 \end{bmatrix}. \tag{A-4}$$

This model is not identifiable and unobservable. Therefore we rewrite the model into a form which is identifiable and observable. The constant μ_s can be included in the long-term variation state $\chi_{s,t}$. We can express the long-term variation of output 2 (median current of nearby track circuits on the same track) as the long-term variation of output 1 plus a term accounting for the difference between the long-term variation in output 2 and track circuit 1. Similarly we can do this for track circuit 3 (median of current of nearby track circuits located on a different track). This gives:

$$y_{1,t} = \chi_{1,t} + ... \tag{A-5}$$

$$y_{2,t} = \chi_{1,t} + \chi_{2,t} + ... \tag{A-6}$$

$$y_{3,t} = \chi_{1,t} + \chi_{3,t} + ... \tag{A-7}$$

Furthermore we can fix the value of a_1 to 1, since we could multiply the state β_t with $\frac{1}{a_1}$. Similarly, we can set b_1 to 1.

This gives us the following state transition equations

$$x_{t+1} = Ax_t + Bu_t + v_t =$$

$$\begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_{t+1} = \begin{bmatrix} g & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos(\omega\tau) & \frac{\sin(\omega\tau)}{\omega} \\ 0 & 0 & 0 & 0 & -\omega\sin(\omega\tau) & \cos(\omega\tau) \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_t + \begin{bmatrix} \zeta \\ \nu_1 \\ \nu_2 \\ \nu_3 \\ 0 \\ 0 \end{bmatrix}_t. \tag{A-8}$$

With the following observation equations:

$$\begin{aligned} y_t &= Cx_t + w_t = \\ \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t &= \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ a_2 & 1 & 1 & 0 & b_2 & 0 \\ a_3 & 1 & 0 & 1 & b_3 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} \eta_{1,t} + c_1\epsilon_t \\ \eta_{2,t} + c_2\epsilon_t \\ \eta_{3,t} + c_3\epsilon_t \end{bmatrix}. \end{aligned} \tag{A-9}$$

The noise covariance matrices are defined with the following coefficients:

$$E\left[v_t v_t^\top\right] = \begin{bmatrix} \sigma_\zeta^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\nu,1}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\nu,2}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\nu,3}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A-10}$$

$$E\left[w_t w_t^\top\right] = \begin{bmatrix} \sigma_{1,1}^2 & \sigma_{2,1} & \sigma_{3,1} \\ \sigma_{2,1} & \sigma_{2,2}^2 & \sigma_{3,2} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma_{3,3}^2 \end{bmatrix}. \tag{A-11}$$

We observe that the model for healthy system behavior consists of six states.

A-2 Adding a fault model to the general model

Our fault models have two states in general. For example:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t + v_t. \tag{A-12}$$

Here, $x_{1,t}$ represent the absolute fault intensity and $x_{2,t}$ is the degradation rate (the gradient of the fault intensity). If we would combine our models by adding the state $x_{1,t}$ to the output y_t, this would make the model unobservable, since the long-term current level $\chi_{s,t}$ cannot be distinguished from the fault $x_{1,t}$. We model the degradation rate $x_{2,t}$ as an effect on the long-term current level $\chi_{s,t}$. This leads to the following state transition equations:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + v_t = \\ \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \\ x_2 \end{bmatrix}_t &= \begin{bmatrix} g & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & h \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & \cos(\omega\tau) & \frac{\sin(\omega\tau)}{\omega} & 0 \\ 0 & 0 & 0 & 0 & -\omega\sin(\omega\tau) & \cos(\omega\tau) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & l \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \\ x_2 \end{bmatrix}_t + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_t + v_t \end{aligned} \tag{A-13}$$

The observation equations are defined by:

$$
\begin{aligned}
y_t &= Cx_t + w_t = \\
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t &= \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ a_2 & 1 & 1 & 0 & b_2 & 0 & 0 \\ a_3 & 1 & 0 & 1 & b_3 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \chi_1 \\ \chi_2 \\ \chi_3 \\ \lambda \\ \dot{\lambda} \end{bmatrix}_t + \begin{bmatrix} \eta_{1,t} + c_1\epsilon_t \\ \eta_{2,t} + c_2\epsilon_t \\ \eta_{3,t} + c_3\epsilon_t \end{bmatrix}.
\end{aligned} \tag{A-14}
$$

The value of h depends on the spatial dependencies of the fault (based on the C matrices derived in Section 4-3-2). When the fault only influences one track circuit (D_1) $h = -1$, while when the fault influences all track circuits on the same track (D_2) $h = 0$. The value of l depends on the type of degradation behavior: l is 0 for constant degradation behavior (after a shock due to abrupt faults), 1 for linear degradation behavior (L) and f for exponential degradation behavior (E).

Appendix B

Additional analyses

B-1 Computational complexity

We have analyzed the computational complexity, by testing the computation time required to run the multiple-model filter using a certain number of models. We have tested the performance for multiple model filtering using one up till 13 models. In Figure B-1 we see that a small quadratic trend is visible: doubling the number, leads to slightly more than double the computation time. This means the computational complexity is $\mathcal{O}(M^2)$ in terms of the number of models used.

B-2 The influence of the number of runs used in the Monte Carlo approach on the prediction

We analyzed how many runs should be used for our Monte Carlo predictions. We made predictions using different number of runs and compared the confidence intervals. In Figure B-2 we have presented the results. We see that the 95 % confidence interval is very erratic when less then $10^2 = 100$ runs are used. With more than 10^2 runs the confidence interval is stabilized. We select 1000 runs as a compromise between accuracy and the amount of computation time required.

B-3 Prognostic performance with Q instead of Q'

We have made predictions using a process noise matrix that was a scaled version of the original process noise matrix Q. We have also tested the prognostic performance when the original matrix Q would have been used. These results are presented in Table B-1.

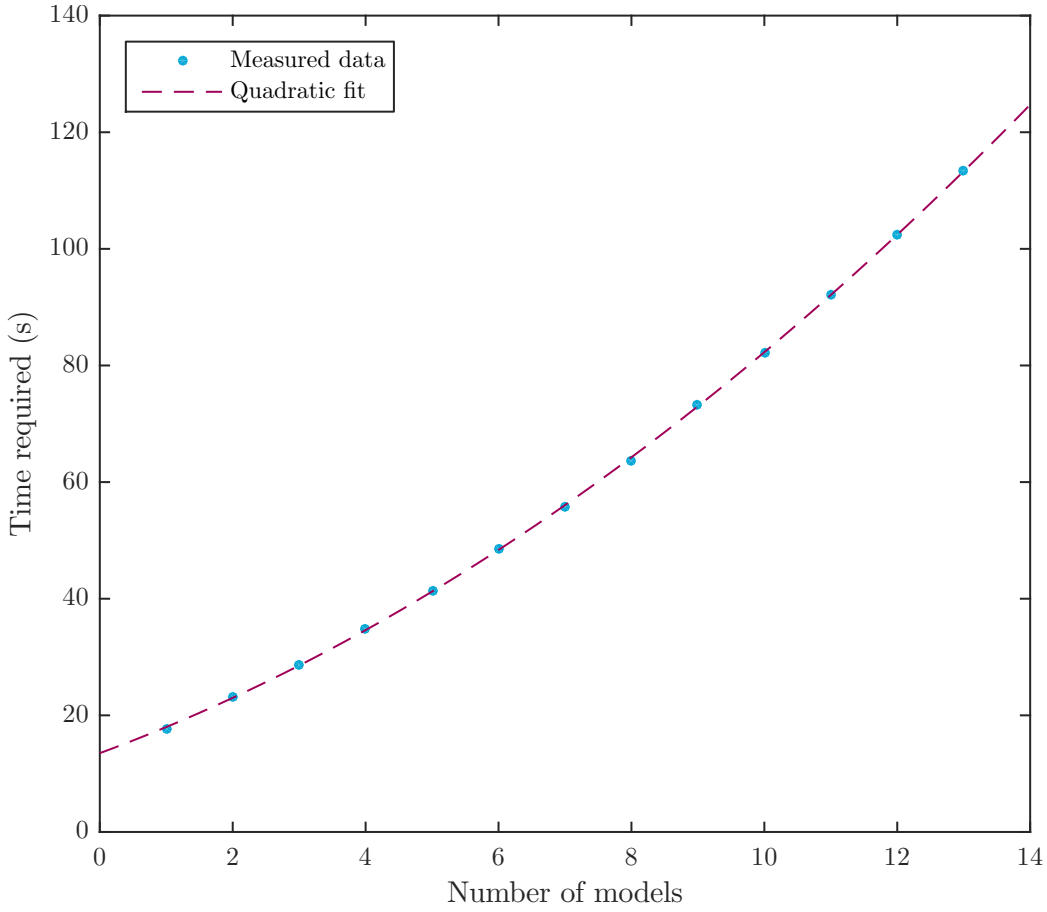


Figure B-1: The influence of adding extra models on the required computation time.

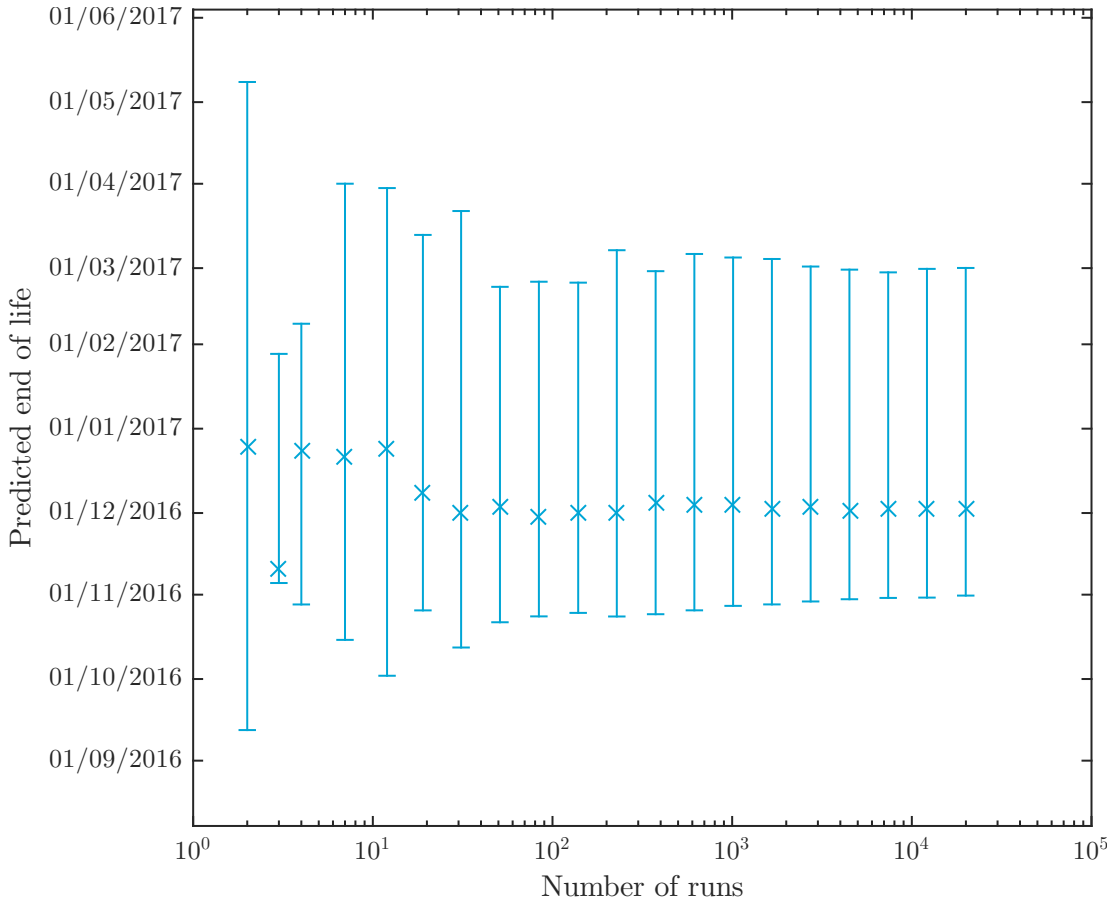


Figure B-2: The influence of the number of runs on the prediction.

Table B-1: Prognostic performance based on 3156 predictions for the remaining useful life in total.

(a) Absolute prediction error.

Fault type	Absolute prediction error (days)	Absolute prediction error first prediction (days)	Absolute prediction error last prediction (days)
Insulated joint defect	3.21	10.79	1.19
Mechanical defect	1.71	9.99	0.15
Ballast degradation	16.06	44.27	6.88
Overall	**6.97**	**21.62**	**2.73**

(b) Prediction accuracy and width 95 % confidence interval.

Fault type	Prediction accuracy	Width 95 % confidence interval first prediction (days)	Width 95 % confidence interval last prediction (days)
Insulated joint defect	82 %	62.00	4.75
Mechanical defect	82 %	66.65	0.46
Ballast degradation	80 %	161.52	39.37
Overall	**81 %**	**69.53**	**14.79**

Appendix C

Fault classification comparison set-up

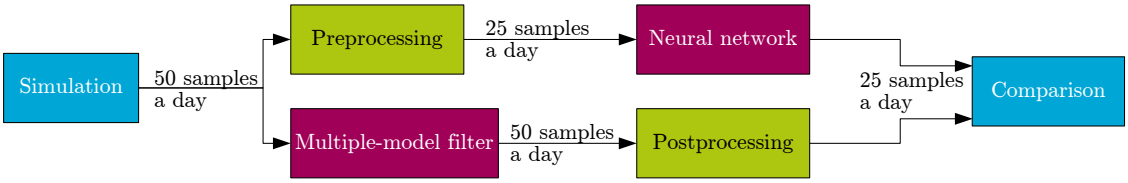


Figure C-1: Fault classification comparison set-up including sampling frequencies.

Bibliography

[1] G. Ackerson and K. Fu, "On state estimation in switching environments," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 10–17, 1970.

[2] R. Ahmad and S. Kamaruddin, "An overview of time-based and condition-based maintenance in industrial application," *Computers and Industrial Engineering*, vol. 63, no. 1, pp. 135–149, 2012.

[3] D. An, N. H. Kim, and J.-H. Choi, "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews," *Reliability Engineering & System Safety*, vol. 133, no. January 2015, pp. 223–236, 2015.

[4] P. Baruah and R. B. Chinnam, "HMMs for diagnostics and prognostics in machining processes," *International Journal of Production Research*, vol. 43, no. 6, pp. 1275–1293, 2005.

[5] T. D. Batzel and D. C. Swanson, "Prognostic health management of aircraft power generators," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, pp. 473–482, 2009.

[6] T. Biagetti and E. Sciubba, "Automatic diagnostics and prognostics of energy conversion processes via knowledge-based systems," *Energy*, vol. 29, no. 12-15, pp. 2553–2572, 2004.

[7] H. A. P. Blom and Y. Bar-Shalom, "Interacting multiple model algorithm for systems with Markovian switching coefficients." *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[8] British Broadcasting Corporation, "Horizon: The pleasure of finding things out," London, Nov. 23, 1981.

[9] C. Bunks, D. McCarthy, and T. Al-Ani, "Condition-based maintenance of machines using hidden Markov models," *Mechanical Systems and Signal Processing*, vol. 14, no. 4, pp. 597–612, 2000.

[10] C. Byington, M. Watson, and P. Stoelting, "A model-based approach to prognostics and health management for flight control actuators," in *Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, 2004, pp. 3551–3562.

[11] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, no. 3, pp. 418–425, 1978.

[12] J. Chen, C. Roberts, and P. Weston, "Fault detection and diagnosis for railway track circuits using neuro-fuzzy systems," *Control Engineering Practice*, vol. 16, no. 5, pp. 585–596, 2008.

[13] Z. L. Cherfi, L. Oukhellou, E. Côme, T. Denoeux, and P. Aknin, "Partially supervised Independent Factor Analysis using soft labels elicited from multiple experts: Application to railway track circuit diagnosis," *Soft Computing*, vol. 16, no. 5, pp. 741–754, 2011.

[14] R. B. Chinnam, "On-line reliability estimation for individual components using statistical degradation signal models," *Quality and Reliability Engineering International*, vol. 18, no. 1, pp. 53–73, 2002.

[15] A. Christer, W. Wang, and J. Sharp, "A state space condition monitoring model for furnace erosion prediction and replacement," *European Journal of Operational Research*, vol. 101, no. 1, pp. 1–14, 1997.

[16] O. L. V. Costa, "Linear minimum mean square error estimation for discrete-time Markovian jump linear systems," *IEEE Transactions on Automatic Control*, vol. 39, no. 8, pp. 1685–1689, 1994.

[17] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-Time Markov Jump Linear Systems*. Springer-Verlag London, 2005.

[18] A. Davies, *Handbook of Condition Monitoring: Techniques and Methodology*. Springer Netherlands, 2012.

[19] T. de Bruin, "Neural network based condition monitoring for track circuits," Msc. Thesis, Delft University of Technology, 2015.

[20] A. Debiolles, L. Oukhellou, and P. Aknin, "Combined use of partial least squares regression and neural network for diagnosis tasks," in *Proceedings of the 17th International Conference on Pattern Recognition*, Los Alamitos, California, Aug. 2004, pp. 573–576.

[21] D. Galar, U. Kumar, R. Villarejo, and C. A. Johansson, "Hybrid prognosis for railway health assessment: An information fusion approach for PHM deployment," *Chemical Engineering Transactions*, vol. 33, pp. 769–774, 2013.

[22] S. Goumas, M. Zervakis, and G. Stavrakakis, "Classification of washing machines vibration signals using discrete wavelet analysis for feature extraction," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 3, pp. 497–508, 2002.

[23] E. Greenberg, *Introduction to Bayesian Econometrics*. Cambridge University Press, 2012.

[24] F. Gustafsson, *Adaptive Filtering and Change Detection*. John Wiley & Sons, Ltd., 2000.

[25] J. Hartikainen, A. Solin, and S. Särkkä, "Optimal filtering with Kalman filters and smoothers: A manual for the Matlab toolbox EKF/UKF," Aalto University, Tech. Rep., 2007.

[26] Inspectie Leefomgeving en Transport, "Quick scan beheer onderhoud hoofdspoorweginfrastructuur ProRail," in Dutch, Tech. Rep., 2012.

[27] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, no. 7, pp. 1483–1510, 2006.

[28] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[29] J. Kittler, M. Hater, and R. P. W. Duin, "Combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 3, pp. 226–239, 1998.

[30] J. Kole, "Zoef zoef," 2009, licensed under CC BY-NC 2.0.

[31] R. Kothamasu, S. H. Huang, and W. H. VerDuin, "System health monitoring and prognostics: a review of current paradigms and practices," *The International Journal of Advanced Manufacturing Technology*, vol. 28, no. 9-10, pp. 1012–1024, 2006.

[32] H. M. Krolzig, M. Marcellino, and G. E. Mizon, "A Markov-switching vector equilibrium correction model of the UK labour market," *Empirical Economics*, vol. 27, no. 2, pp. 233–254, 2002.

[33] X. R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *IEEE Transactions on Automatic Control*, vol. 41, no. 4, pp. 478–493, 1996.

[34] D. Lin, D. Banjevic, and A. K. S. Jardine, "Using principal components in a proportional hazards model with applications in condition-based maintenance," *Journal of the Operational Research Society*, vol. 57, no. 8, pp. 910–919, 2005.

[35] C. J. Lu and W. O. Meeker, "Using degradation measures to estimate a time-to-failure distribution," *Technometrics*, vol. 35, no. 2, pp. 161–174, 1993.

[36] J. Luo, M. Namburu, K. Pattipati, L. Qiao, M. Kawamoto, and S. Chigusa, "Model-based prognostic techniques," in *Proceedings of the 2003 IEEE Systems Readiness Technology Conference*, Anaheim, California, Sep. 2003, pp. 330–340.

[37] J. Luo, K. Pattipati, L. Qiao, and S. Chigusa, "Model-based prognostic techniques applied to a suspension system," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 38, no. 5, pp. 1156–1168, 2008.

[38] D. T. Magil, "Optimal Adaptive Estimation of Sampled Stochastic Processes," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 434–439, 1965.

[39] W. Mansveld, "Brief van de Staatssecretaris van Infrastructuur en Milieu," in *Kamerstuk van de Tweede Kamer der Staten-Generaal 29984*, no. 579, Den Haag, In Dutch, Feb. 2015.

[40] H. Monsef, A. Ranjbar, and S. Jadid, "Fuzzy rule-based expert system for power system fault diagnosis," *IEE Proceedings - Generation, Transmission and Distribution*, vol. 144, no. 2, p. 186, 1997.

[41] E. Moolman, "A Markov switching regime model of the South African business cycle," *Economic Modelling*, vol. 21, pp. 631–646, 2004.

[42] K. P. Murphy, "Dynamic Bayesian networks: Representation, inference and learning," Ph.D. dissertation, University of California, Berkeley, 2002.

[43] NRC Handelsblad, "NS voelt zich overvallen door plotselinge wisselstoring," In Dutch, Feb. 20, 2014.

[44] M. Orchard, F. Tobar, and G. Vachtsevanos, "Outer feedback correction loops in particle filtering-based prognostic algorithms: Statistical performance comparison," *Studies in Informatics and Control*, vol. 18, no. 4, pp. 295–304, 2009.

[45] L. Oukhellou, A. Debiolles, T. Denoeux, and P. Aknin, "Fault diagnosis in railway track circuits using Dempster-Shafer classifier fusion," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 1, pp. 117–128, 2010.

[46] G. Palem, "Condition-based maintenance using sensor arrays and telematics," *International Journal of Mobile Network Communications & Telematics*, vol. 3, no. 3, pp. 19–28, 2013.

[47] E. Parzen, "On estimation of a probability density function and mode," *Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[48] A. P. Patra and U. Kumar, "Availability analysis of railway track circuits," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 224, no. 3, pp. 169–177, 2010.

[49] R. J. Patton and J. Chen, "Observer-based fault detection and isolation: Robustness and applications," *Control Engineering Practice*, vol. 5, no. 5, pp. 671–682, 1997.

[50] B. Paya, I. Esat, and M. Badi, "Artificial neural network based fault diagnostics of rotating machinery using wavelet transforms as a preprocessor," *Mechanical Systems and Signal Processing*, vol. 11, no. 5, pp. 751–765, 1997.

[51] C.-Y. Peng and S.-T. Tseng, "Mis-specification analysis of linear degradation models," *IEEE Transactions on Reliability*, vol. 58, no. 3, pp. 444–455, 2009.

[52] Y. Peng, M. Dong, and M. J. Zuo, "Current status of machine prognostics in condition-based maintenance: A review," *The International Journal of Advanced Manufacturing Technology*, vol. 50, no. 1-4, pp. 297–313, 2010.

[53] E. Phelps, P. Willett, T. Kirubarajan, and C. Brideau, "Predicting time to failure using the IMM and excitable tests," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 37, no. 5, pp. 630–642, 2007.

[54] ProRail, "Onderhoudsdocument: Laagfrequent spoorstroomlopen 75 Hz secties in ET-gebied," in Dutch, Tech. Rep., 2013.

[55] F. R. Redeker, "POSS: Railway condition monitoring developed by a maintainer," in *Proceedings of the 5th IET Conference on Railway Condition Monitoring and Non-Destructive Testing*, Derby, United Kingdom, Nov. 2011.

[56] L. C. K. Reuben and D. Mba, "Diagnostics and prognostics using switching Kalman filters," *Structural Health Monitoring*, vol. 13, no. 3, pp. 296–306, 2014.

[57] C. Roberts, H. Dassanayake, N. Lehrasab, and C. Goodman, "Distributed quantitative and qualitative fault diagnosis: Railway junction case study," *Control Engineering Practice*, vol. 10, no. 4, pp. 419–429, 2002.

[58] X. Rong Li and V. P. Jilkov, "A survey of maneuvering target tracking - Part V: Multiple-model methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1255–1321, 2005.

[59] X. Rong Li and Y. Zhang, "Numerically robust implementation of multiple-model algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 1, pp. 266–278, 2000.

[60] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

[61] S. M. Ross, *Introduction to Probability Models*. Academic Press, 2007.

[62] B. Samanta and K. Al-Balushi, "Artificial neural network based fault diagnostics of rolling element bearings using time-domain features," *Mechanical Systems and Signal Processing*, vol. 17, no. 2, pp. 317–328, 2003.

[63] M. Sandidzadeh and M. Dehghani, "Intelligent condition monitoring of railway signaling in train detection subsystems," *Journal of Intelligent and Fuzzy Systems*, vol. 24, no. 4, pp. 859–869, 2013.

[64] A. Saxena, J. Celaya, and B. Saha, "Metrics for offline evaluation of prognostic performance," *International Journal of Prognostics and Health Management*, vol. 1, no. 1, pp. 1–20, 2010.

[65] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation: A review on the statistical data driven approaches," *European Journal of Operational Research*, vol. 213, no. 1, pp. 1–14, 2011.

[66] J. Z. Sikorska, M. Hodkiewicz, and L. Ma, "Prognostic modelling options for remaining useful life estimation by industry," *Mechanical Systems and Signal Processing*, vol. 25, no. 5, pp. 1803–1836, 2011.

[67] H. Sohn, K. Worden, and C. R. Farrar, "Statistical damage classification under changing environmental and operational conditions," *Journal of Intelligent Material Systems and Structures*, vol. 13, no. 9, pp. 561–574, 2002.

[68] A. V. H. Steenkamp, "Spoorstroomlopen," Nederlandse Spoorwegen, in Dutch, Tech. Rep., 1981.

[69] L. Tang, DeCastro, G. Jacprzynski, K. Goebel, and G. Vachtsevanos, "Filtering and prediction techniques for model-based prognosis and uncertainty management," in *Proceedings of the 2010 Prognostics & System Health Management Conference*, Macau, China, 2010, pp. 186–195.

[70] G. Theeg, E. Anders, and S. V. Vlasenko, *Railway Signalling & Interlocking: International Compendium.* Eurailpress, 2009.

[71] A. Timmermann, "Forecast Combinations," in *Handbook of Economic Forecasting.* Elsevier B.V., 2006, pp. 135–196.

[72] N. Tudoroiu and K. Khorasani, "Satellite fault diagnosis using a bank of interacting Kalman filters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1334–1350, 2007.

[73] J. K. Tugnait and A. H. Haddad, "A detection-estimation scheme for state estimation in switching environments," *Automatica*, vol. 15, no. 4, pp. 477–481, 1979.

[74] J. K. Tugnait, "Comments on "State estimation for discrete systems with switching parameters"," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 3, p. 464, 1979.

[75] J. M. van Noortwijk, "A survey of the application of gamma processes in maintenance," *Reliability Engineering & System Safety*, vol. 94, no. 1, pp. 2–21, 2009.

[76] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311, 2003.

[77] ——, "A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 313–326, 2003.

[78] ——, "A review of process fault detection and diagnosis: Part III: Process history based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 327–346, 2003.

[79] K. Verbert, B. De Schutter, and R. Babuška, "Exploiting spatial and temporal dependencies to enhance fault diagnosis: Application to railway track circuits," in *Proceedings of the 2015 European Control Conference*, Linz, Austria, Jul. 2015, pp. 3052–3057.

[80] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[81] X. Wang and V. L. Syrmos, "Interacting multiple particle filters for fault diagnosis of non-linear stochastic systems," in *Proceedings of the 2008 American Control Conference*, Seattle, Washington, 2008, pp. 4274–4279.

[82] S. Zhang and R. Ganesan, "Multivariable trend analysis using neural networks for intelligent diagnostics of rotating machinery," *Journal of Engineering for Gas Turbines and Power*, vol. 119, no. 2, p. 378, 1997.

[83] Y. Zhang and J. Jiang, "An interacting multiple-model based fault detection, diagnosis and fault-tolerant control approach," in *Proceedings of the 38th IEEE Conference on Decision & Control*, Phoenix, Arizona, Dec. 1999, pp. 3593–3598.

[84] Y. Zhang and X. Rong Li, "Detection and diagnosis of sensor and actuator failures using interacting multiple-model estimator," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, 1998, pp. 1293–1313.

[85] F. Zhao, J. Chen, L. Guo, and X. Li, "Neuro-fuzzy based condition prediction of bearing health," *Journal of Vibration and Control*, vol. 15, no. 7, pp. 1079–1091, 2009.

[86] R. R. Zhou, N. Serban, and N. Gebraeel, "Degradation modeling applied to residual lifetime prediction using functional data analysis," *The Annals of Applied Statistics*, vol. 5, no. 2B, pp. 1586–1610, 2011.

[87] E. Zio, "Reliability engineering: Old problems and new challenges," *Reliability Engineering & System Safety*, vol. 94, no. 2, pp. 125–141, 2009.

Glossary

List of Symbols

α_1	Maximum current of a healthy system when the section is occupied (A)
α_2	Minimum current of a healthy system when the section is clear (A)
β_t	Wetness of the ballast (V)
$\chi_{s,t}$	Output-specific causes of long-term variation (A)
δ	Process noise scaling factor for predictions
ϵ_t	Unknown common causes of variation in the measured currents
$\eta_{s,t}$	Unknown output-specific causes of variation in the measured current (A)
γ_1	Maximum current required to switch signal from clear to occupied (A)
γ_2	Minimum current required to switch signal from occupied to clear (A)
κ_t	Innovation of long-term variation affecting all outputs
λ_t	Periodic effects
μ_s	Nominal current (A)
$\nu_{s,t}$	Output-specific long-term variation innovation
π	Transition probability
π_{ij}	Transition probability from mode i to mode j
σ^2	Variance
τ	Sampling time
θ	Active mode of the system
ξ_t	Causes of long-term variation affecting all outputs
ζ_t	Measurement noise on the rain sensor (V)
$\hat{T}$	Predicted remaining useful life
$\mathcal{A}$	Relative accuracy
$\mathcal{L}$	Logarithmic loss likelihood
A	State transition matrix of a state-space system
a_s	Coefficient denoting the influence of wetness of the ballast on the measured current (A/V)

B	Input matrix of a state-space system
b_s	Coefficient denoting the influence of periodic effects on the measured current (A)
C	Output matrix of a state-space system
c_s	Coefficient denoting the influence of common causes of variation on the output $y_{s,t}$ (A)
D	Feedthrough matrix of a state-space system
d_s	Coefficient denoting the influence of common long-term variation causes on the measured current $y_{s,t}$ (A)
f_θ	Pole location of exponential degradation behavior for fault mode θ
h	Parameter used in observable realization which depends on the degradation behavior of the fault
I_t	Current flowing through the relay of the track circuit (A)
$I_{i,t}$	Track circuit measurement for track circuit i (A)
k	Number of time steps ago for which hypotheses having a different history are merged in the generalized pseudo Bayes algorithm
l	Parameter used in observable realization which depends on the spatial dependencies of the fault
M	The number of models
N	Number of hypotheses contained in the N best strategy
Q	Process noise variance-covariance matrix
R	Measurement noise variance-covariance matrix
T	Actual remaining useful life
t	Time step
u_t	System input / rain sensor measurements (V)
v_t	Vector with process noise in a state-space system
w_t	Vector with measurement noise in a state-space system
x_t	Vector with states in a state-space system
$x_{i,t}$	State i in a state-space system
y_t	Vector with outputs / preprocessed track circuit current measurements (A)
$y_{s,t}$	System output s (A)
A	Abrupt degradation behavior
D_1	The fault influences one specific section.
D_2	The fault influences track circuits on the same track.
D_3	The fault influences track circuits along the path of a specific train
D_4	The fault influences all nearby sections
E	Exponential degradation behavior
I	Intermittent degradation behavior
L	Linear degradation behavior
$_\epsilon$	Subscript denoting unknown output-specific variation
$_\eta$	Subscript denoting unknown short-term variation affecting all outputs

κ	Subscript denoting the innovation for long-term variation affecting all outputs
ν	Subscript denoting the innovation for output-specific long-term variation
degradation	Subscript denoting a model describing degradation behavior
healthy	Subscript denoting the healthy model
step	Subscript denoting a model describing a step in one of the states
ζ	Subscript denoting measurement noise on the rain sensor
s	Subscript denoting the output index
t	Subscript denoting the time index
$+$	Superscript denoting the mixed state estimate used as input for Kalman filter