Hidden semi-Markov Models for Predictive Maintenance of Rotating Elements

Vom Fachbereich Maschinenbau an der
Technischen Universität Darmstadt
zur
Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.) genehmigte

Dissertation

vorgelegt von

Christoph Anger M. Sc.

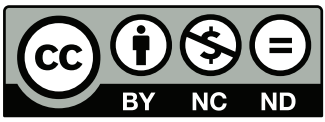
aus Frankfurt am Main

Berichterstatter: Prof. Dr.-Ing. Uwe Klingauf
Mitberichterstatter: Prof. Dr.-Ing. Tobias Melz

Tag der Einreichung: 09. November 2017
Tag der mündlichen Prüfung: 20. Februar 2018

Darmstadt 2017

D 17



Abstract

The dissertation at hand introduces a novel algorithm to predict the remaining useful life (RUL) of rotating components such as grooved ball bearings or gear boxes. The focus of the implemented method is on so-called Hidden semi-Markov Models (HsMM), which are suitable for the modeling of sequential events. The selected method is designed to support maintenance processes based on advisory generation in the context of predictive maintenance. In this regard, the goal of predictive maintenance is the generation of predictions about the RUL of examined components based on their current state. Thus, maintenance events can be scheduled more precisely, downtime is reduced, and the actual useful life of components can be exploited.

After an initial classification of the proposed method in the context of Prognostics and Health Management (PHM), the concept is described. The algorithm is based on methods, which apply historical data of the component's degradation process for the estimation of the current and future damage state. A novel concept in the field of HsMM permits the identification of similar damage states within different degradation datasets to obtain more information about the damage process of the examined components. A further research question analyzes, whether the consideration of available information about the component's endured load increases the prognostic performance. First results in the context of verification conclude the concept description.

Subsequently, the design and realization of a new test rig, which permits an accelerated bearing aging for induction machines due to a so-called bearing current, is presented. Here, an alternating current flows through the tested bearing and reduces its life cycle significantly. By means of these data, the concept is evaluated. In comparison to state-of-the-art methods in field of bearing PHM, the validation is executed by examining the motor current of the induction machine instead of the widespread analysis of the resulting vibration signal. The results indicate that the precise and accurate prediction of the bearing's RUL is possible. In addition, the consideration of already endured load for the generation of life cycle predictions is beneficial. A selected state-of-the-art algorithm, also based on HsMM, permits a realistic evaluation of the achieved prognostic performance.

The dissertation ends with the estimation of possible cost savings in an exemplary aircraft maintenance scenario. For this, the obtained prognostic results are assessed with a state-of-the-art cost-benefit analysis tool. The outcome indicates

that the application of the proposed algorithm leads to savings due to e.g. decreased downtimes.

Kurzfassung

Die vorliegende Dissertation stellt einen neuartigen Algorithmus zur Lebensdauerbestimmung von rotierenden Bauteilen wie Rillenkugellagern oder Getrieben vor. Im Zentrum der implementierten Methode stehen sogenannte Hidden semi-Markov Modelle (HsMM), die sich für die Abbildung von sequentiellen Ereignissen eignen. Die gewählte Methode soll den Wartungsbetrieb durch Handlungsempfehlungen im Rahmen der prädiktiven Instandhaltung unterstützen. Ziel der prädiktiven Instandhaltung ist, ausgehend von dem aktuellen Zustand der Komponente, Prognosen hinsichtlich der verbleibenden Restlebensdauer zu generieren. Dadurch können Wartungsevents planbarer gestaltet, Stillstandszeiten verringert und der nutzbare Lebenszyklus von Komponenten ausgenutzt werden.

Nach einer Einordung der Methode in den Kontext des Prognostics and Health Managements, erfolgt die Konzeptbeschreibung. Der Algorithmus besteht dabei aus Verfahren, die, basierend auf historischen Daten über den Degradierungsprozess der untersuchten Komponente, den aktuellen Schadenszustand schätzen sowie den zukünftigen Schadensverlauf prognostizieren. Ein neuartiges Konzept ermöglicht es, mithilfe der HsMM vergleichbare Schadenszustände in verschiedenen Degradierungsdaten zu identifizieren, um somit mehr Informationen über den Schädigungsprozess der untersuchten Komponente zu erhalten. Eine weitere Forschungsfrage betrachtet, ob das Einbinden von verfügbaren Informationen über die Belastung der Komponente Vorteile hinsichtlich der Prognosegenauigkeit zeigt. Erste Ergebnisse im Rahmen einer Verifikation schließen die Konzeptbeschreibung ab.

Nachfolgend wird die Auslegung und Umsetzung eines Prüfstands beschrieben, der eine künstlich beschleunigte Rillenkugellageralterung in Asynchronmotoren mithilfe von sogenanntem Lagerstrom realisiert. Hierbei fließt ein elektrischer Wechselstrom durch das Testlager und reduziert drastisch seine Lebensdauer. Mit diesen Daten wird im Anschluss das Konzept validiert. Im Vergleich zu anderen Verfahren aus dem Bereich der Lagerlebensdauerschätzung, die den Körperschall für die Zustandsüberwachung untersuchen, wird für die Validierung in dieser Dissertation der Statorstrom des Asynchronmotors genutzt. Die Ergebnisse zeigen, dass auch hiermit eine genaue Prognose der Restlebensdauer von Kugellagern möglich ist. Auch die Verwendung der bereits ertragenen Belastung in die Generierung von Prognosen weisen deutliche Vorteile auf. Ein etabliertes Vergleichskonzept,

welches ebenfalls auf HsMM basiert, ermöglicht eine realistische Einordnung der Prognoseergebnisse.

Abschließend werden die Resultate hinsichtlich möglicher Einsparungen in einem exemplarischen Flugzeugwartungsszenario untersucht. Hierfür dient ein Kosten-Nutzen-Rechnung-Simulationswerkzeug aus der Literatur. Es zeigt sich, dass durch die Anwendung des vorgestellten Algorithmus Kosteneinsparungen, insbesondere im Bereich der Stillstandfolgekosten, möglich sind.

Danksagung

Die vorliegende Dissertation wurde während meiner fünfjährigen Zeit als wissenschaftlicher Mitarbeiter am Institut für Flugsysteme und Regelungstechnik der Technischen Universität Darmstadt angefertigt. Besonderer Dank gilt dem Leiter des Instituts, *Professor Dr.-Ing. Uwe Klingauf*, für die fachliche Unterstützung, die schöne Zeit am Institut und die produktive Zusammenarbeit in der Vorlesungsbetreuung, bei der ich immer etwas für mich mitnehmen konnte. *Professor Dr.-Ing. Tobias Melz* danke ich für die Übernahme des Korreferats und das Interesse an meinem Dissertationsthema.

Bedanken möchte ich mich zudem bei allen Kolleginnen und Kollegen, die mich in dieser Zeit am Institut begleitet haben. Durch diverse Aktivitäten in und abseits der Universität sind dabei Freundschaften auch über etwaige Teamgrenzen hinaus entstanden. Besonders danke ich hier *Christian Preusche* und *Alexander Kählert*, die unter anderem durch fachliche Diskussionen über die Aspekte der Prognose und Diagnose von technischen Systemen dafür sorgten, dass diese Zeit sehr abwechslungsreich und kurzweilig war.

Bei *Simon Mehringskötter* bedanke ich mich für das kritische Auseinandersetzen mit meiner Dissertation und für die Anregungen, die hierdurch entstanden sind. Auch den Studierenden, die durch Abschlussarbeiten und Studienprojekte zu dieser Dissertation beigetragen und meinen Horizont erweitert haben, möchte ich danken. Insbesondere *Tim Schickel* danke ich für die Unterstützung im Aufbau des Prüfstands während seiner Masterthesis.

Zuletzt danke ich meiner Familie und meinen Freunden, die ich in der Schreibphase oft vertrösten musste. Meinen Eltern und besonders meiner *Mutter Elke* danke ich, dass sie mich immer unterstützt haben. Meiner Freundin *Katja Hein* danke ich für die vielen Ratschläge und den Halt auch in schwierigen Phasen.

Darmstadt, im Juni 2018

Christoph Anger

Contents

Symbols and abbreviations

Latin letters

$\boldsymbol{A}$, a_{ij}	Transition matrix
$\boldsymbol{B}$, b_{jk}	Emission matrix
BIC	Bayes Information Criterion
$\boldsymbol{C}$	Clusters, damage classes
ΔC	Cost benefits (Predictive vs. corrective maintenance)
C_P	Costs for implementation of predictive maintenance
$\boldsymbol{D}$, $\boldsymbol{d}_i$	Duration matrix
f	Frequency
I_{ai}	Current of phase a (ith data set)
m_i	Model i
m_{net}	Net model
M	Number of damage classes/cluster
N	Number of damage states
N_p	Number of predictions
N_{td}	Number of training data
n_s	Number of samples
O_t	Observations at time t
q_t	State at time t
r	Predicted RUL
r^*	Correct RUL
$\boldsymbol{S}_C$	Class probability
S_i	Damage state
$\boldsymbol{S}_{YY}$	Amplitudes of power spectral density spectrum
$\hat{\boldsymbol{S}}_{YY}$	Normed amplitudes of power spectral density spectrum
Sil	(Averaged) Silhouette coefficient
$\boldsymbol{T}$	Training output/target vector
t	Time
t_{EoL}	End of Life
t_{FN}	Acceptable prediction error for False Negative
t_{FP}	Acceptable prediction error for False Positive

t_p	Time of first prediction
t_{PS}	Total prediction time span
$\boldsymbol{U}$	System input
$\boldsymbol{u}_\Sigma$	Cumulative load
v_k	kth observation
$\boldsymbol{x}$	State vector, data points
$\boldsymbol{X}$	Training input matrix
$\boldsymbol{Y}$	System output

Indices

$_c$	Continuous feature matrix
$_e$	Supply frequency
$_s$	Characteristic bearing fault frequency in stator current
$_v$	Characteristic bearing fault frequency in vibration

Greek letters

A	New condition damage state
α	Accuracy bound parameter
$\boldsymbol{\gamma}_t$	State probability at time t
Δ	Prediction error
$\boldsymbol{\epsilon}$	Residuum vector
$\boldsymbol{\Theta}$	Parameter vector
λ	Normed life span
$\boldsymbol{\lambda}$	Hidden semi-Markov Model parameters
$\mu_{\epsilon,l}$	Mean prediction error of unit l
ξ	Minimum acceptable probability mass within accuracy bounds
$\boldsymbol{\Phi}$	Feature matrix
$\boldsymbol{\pi}$	Initial state probability
Ω	Failure condition damage state

Mathematical operations

$\sigma(x)$	Standard deviation of signal x

$\mu(x)$	Expected value of signal x
$\overline{x}$	Arithmetic mean of signal x

Abbreviations

AL	α-λ accuracy
ANN	Artificial Neural Network
BIT(E)	Built-In Test (Equipment)
CAD	Computer Aided Design
CBM	Condition Based Maintenance
CDF	Cumulative Distribution Function
CM	Condition Monitoring
EDM	Electrical Discharge Machining
EoL	End of Life
FFT	Fast Fourier Transformation
FMECA	Failure Modes, Effects, and Criticality Analysis
FN	False Negative
FNR	False Negative Rate
FP	False Positive
HMM	Hidden Markov Model
HsMM	Hidden semi-Markov Model
KNN	K Nearest Neighbor
LOOCV	Leave One Out Cross Validation
MAPE	Mean Absolute Percentage Error
MCSA	Motor Current Signature Analysis
MRO	Maintenance, Repair, and Overhaul
NFF	No Fault Found
OSA-CBM	Open System Architecture for Condition Based Maintenance
PDF	Probability Density Function
PH	Prognostic Horizon
PHM	Prognostics and Health Management
PM	Predictive Maintenance
PMF	Probability Mass Function
PSD	Power Spectral Density
RMS	Root Mean Square
RUL	Remaining Useful Life
RUL*	Normed Remaining Useful Life
SFDR	Specific False Discovery Rate

SSD	Sample Standard Deviation
STFT	Short Time Fourier Transformation
SVM	Support Vector Machine
TP	True Positive

1 Introduction

This thesis provides a new approach to the life cycle estimation of rotating components in the field of predictive maintenance. With the aid of this new method, decision-making regarding relevant maintenance events and troubleshooting in the case of upcoming breakdowns should be supported.

This first chapter gives a rough outline of the importance of this topic and the research questions examined in this thesis. Thus, the chapter starts with a brief motivation beginning with the benefits of predictive maintenance and the challenges stemming from its application. The main goals of this thesis are then stated. The final section provides an overview of the structure of the thesis.

1.1 Motivation

> "*Predictive (...) Maintenance is a shining goal for industrial customers and the Industry 4.0 initiative*" *[Pap17]*

Over the recent years, the term *Predictive Maintenance* (PM) appears more frequently in connection with *safety*, *cost-efficiency*, and *sustainability*. Currently, the German Federal Ministry of Economics and Energy has also identified the importance of PM and subsidizes the research and development in this field in the context of the 5th National Civil Aeronautical Research Program [Bun16]. This should ensure the fulfillment of the determined goals on Federal and European levels in terms of effective and efficient aviation or resource conservation [Bun14], [Eur11]. Since *maintenance, repair, and overhaul* (MRO) companies oversee the aircraft for the longest period of its life cycle, they are emphasized in [Bun14] of great importance for the achievement of these objectives.

In general, PM refers to a maintenance concept that schedules maintenance events based on the analysis and forecasting of the current and future system efficiency or mechanical behavior [Mob02, p. 5]. In contrast to state-of-the-art maintenance concepts, which rely on failure statistics, i.e. *preventive* or *predetermined maintenance*, or even allow the appearance of breakdowns, i.e. *corrective maintenance*, PM stands through the use of two inherent steps: *fault diagnosis* and *fault prognosis*. Whereas the task of fault diagnosis comprises the accurate detection, isolation, and identification of faults, which also includes the quantification

of fault severity, the outcome of fault prognosis is the precise prediction of future fault progress and its crossing of predefined failure thresholds [VLR+06, p. 15].

In accordance to [Mob02, p. 5],the benefits of PM are manifold, since it improves:

- Productivity
- Product quality
- Overall effectiveness
- Machinery availability
- Fault cause detection
- Safety

All benefits lead to an assumed reduction of direct and indirect maintenance costs. The International Air Transport Association (IATA) analyzed in [IAT15] the development of total maintenance costs of global MRO companies within the years 2010 to 2014 without the application of PM. One conclusion is that direct maintenance costs increased by 42 % within these five years, whereas in comparison the fleet size grew only by 21 % [IAT15, p. 51]. In total, global MRO companies spent $ 62.1 billion (direct and indirect maintenance costs excluding overheads) in 2014, which corresponds to around 9 % of total operational costs [IAT15, p. 13]. In contrast, potential cost reductions of 15 to 20 % are expected through the development and introduction of new technologies such as aircraft health monitoring systems, big data, and PM [IAT15, p. 15]. A similar conclusion is drawn in the energy sector by the U.S. Department of Energy in [U.S10] by estimating the cost benefits of PM in comparison to state-of-the-art maintenance concepts; according to [U.S10, Section 5.4], a properly functioning PM program can generate savings of up to 12 % in comparison to a predetermined program, and up to 40 % compared to a corrective maintenance approach. However, this benefit comes with the need for skilled staff who has to interpret the prognostics results, as well as an immense upstream investment [U.S10, Section 5.4].

One key enabler for PM is the steadily increasing amount of available sensor data as a result of digitization [Pap17]. Strategies such as *Industry 4.0* or *Industrial Internet of Things* allow the implementation of a great variety of sensors into system networks. In addition, PM also benefits from new sensor developments. In [Hei03] for instance, a rolling-element bearing with integrated strain gauges is presented. With the aid of these sensors, changes in the bearing's components are detected, which permits on-line monitoring of its current condition. However, with an increasing quantity and plurality of sensors, the loss of overview about the available data and consequently the need for *data mining* methods also rises in order to separate significant from unimportant information. In [Pap17] an oil platform is quoted

as an example, where only 1 % of data sampled by 30,000 sensors is considered for condition monitoring.

Thus, during the design phase of new concepts for the support of PM, it is necessary to appraise, whether an additional sensor is necessary, or if the current state can be detected by a combination of already mounted sensors. One promising example of possible sensor reduction is the condition monitoring of induction machines. The state-of-the-art approach to assess the current condition of stator windings, squirrel cage, or bearings is to analyze the vibration signal. New concepts make use of the stator current sensors that are required for the speed control of the induction engine. Whereas the fault diagnosis of induction machines by means of current sensors has been successfully proven, the suitability of this in the context of fault prognosis remains unknown.

1.2 Aims

Based on the above mentioned challenges of predictive maintenance, a new data-based concept to support the *decision-making* and *initiation of maintenance tasks* in the case of rotating elements is presented in the thesis at hand. Both processes are based on the so-called Prognostic and Health Management (PHM) of components, which provides for instance the remaining useful life (RUL) of monitored units and examines the combination of different fault cases. Thus, the proposed algorithm is more precisely located in the field of PHM, since the estimation of RUL is the focus of this thesis. The approach comprises a method to identify a significant set of features which most accurately represents the actual state of a degrading component. The processing of this feature subset requires new approaches in both fault diagnosis and prognosis. Thus, a method to provide comparability between different sets of degrading components with regard to damage states within a complete life cycle will be introduced. In addition, the impact of including the experienced load during the component's use on the prognostic performance is examined. The results are compared to a state-of-the-art method for fault diagnosis and prognosis.

For the verification and validation of this concept, the data of degrading bearings in induction machines are applied. A new test rig design is necessary in order to analyze the stator current as well as to artificially accelerate the degradation process. Finally, the impact of the achieved prognostic performance on current maintenance with regard to costs is evaluated.

1.3 Thesis outline

The structure of this thesis is summarized in Figure 1.1. In addition, the outcome of each particular chapter is summarized in terms of research questions. The thesis is divided into six chapters. After the introductory chapter, contemporary definitions and terms in the field of PHM are presented in Chapter 2. Besides the basic steps of PHM and a method to assess the cost benefits of PHM algorithms, the lack of state-of-the-art approaches is also identified. The fundamentals of the proposed PHM algorithm are then introduced in Chapter 3. The mathematical correctness is proven with a verification at the end of this chapter. In Chapter 4, the selected case study to validate the proposed PHM algorithm is presented. Thus, a new test rig design for the generation of ball bearing degradation data in induction machines is described. Bearing current, i.e. an alternating current that runs through the test bearing, is selected as a method for accelerated bearing aging and is therefore outlined in this chapter. The performance of the proposed algorithm is then determined in terms of a validation. Here, the focus is on the prognostic performance. Based on derived performance indicators, possible economic benefits in the case of an application are analyzed at the end of this chapter. All achieved goals and possible future work are summarized in Chapter 6.

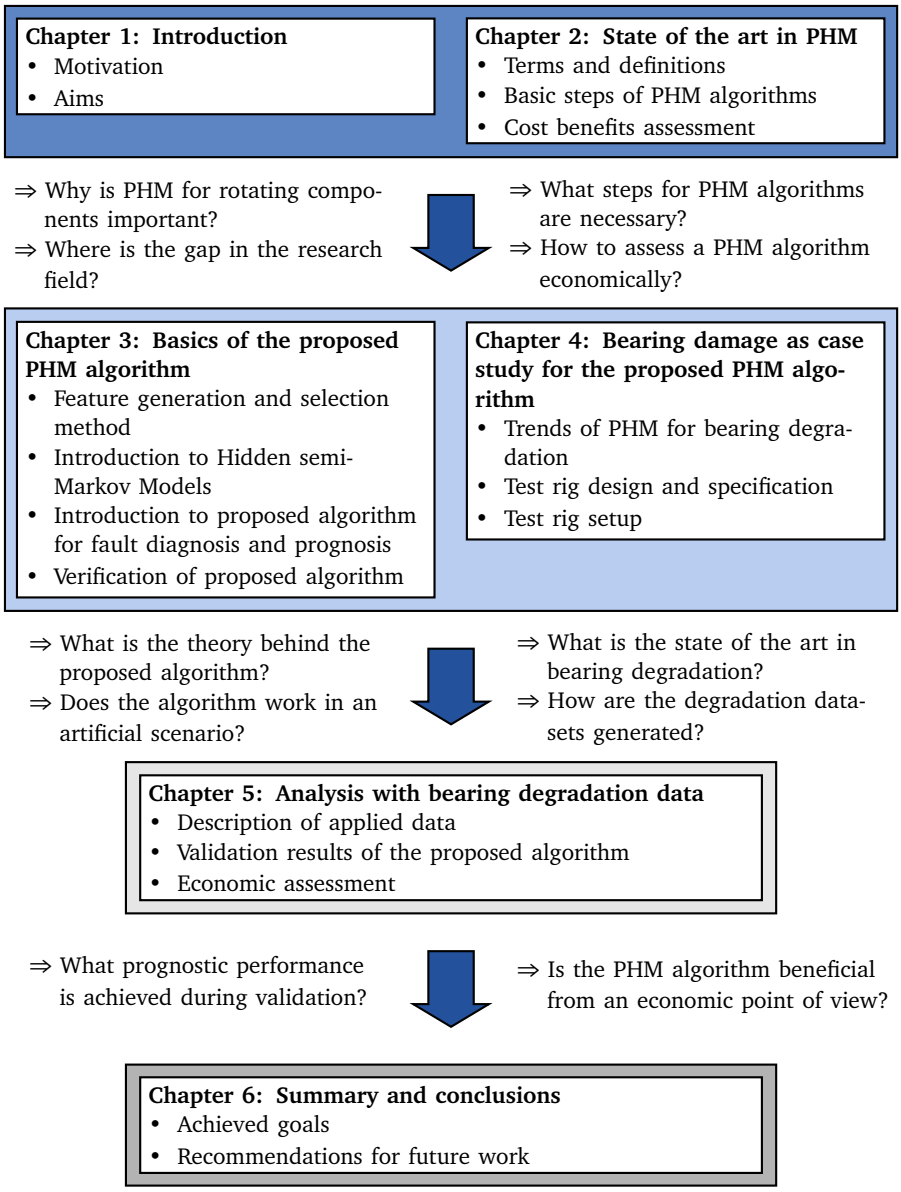


Figure 1.1.: Thesis structure and resulting research questions

2 State of the art in Prognostics and Health Management

The subsequent chapter provides an overview of current Prognostics and Health Management (PHM) approaches. The chapter begins with the introduction of various aspects of industrial maintenance. Different maintenance types and strategies are introduced in this first section. Terms and definitions in the context of PHM as well as its application fields are then presented. Since PHM consists of predefined steps, several state-of-the-art approaches for each module are discussed. To assess the results of the PHM algorithm proposed in this thesis, selected prognosis performance metrics are introduced. In addition, one goal of this thesis is the evaluation of the results with regard to economic benefits. Therefore, the outline of a tool for cost-benefit analysis in aircraft maintenance is presented. The chapter closes with the delimitation of this thesis towards other approaches in the field of PHM.

2.1 Aspects of industrial maintenance

As in many other disciplines, the terminologies and definitions in the area of maintenance are not consistent. Thus, this section provides an overview of the different approaches to classifying state-of-the-art maintenance procedures.

Maintenance is defined by [Deu10, p. 6] as "the combination of all technical, administrative, and managerial actions during the life cycle of an item aimed to retain it in or restore it to a state, in which it can perform the required function". [Deu12] divide this procedure into four aspects as depicted in Figure 2.1.

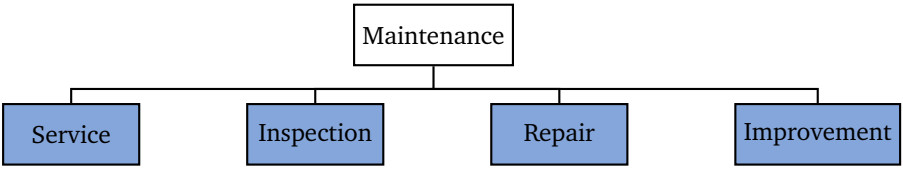


Figure 2.1.: Aspects of maintenance procedures. Based on [Deu12]

Here, *service* involves measures to delay the degradation of *reserve of wear-out* [Deu12]. Reserve of wear-out is defined as a reserve of tolerable function fulfillment under determined conditions (see Figure 2.4). During the *inspection* step, the conformity of an item is examined by measuring, observing, or testing its characteristics. *Repair* comprises all necessary physical actions required to restore the pre-defined function of an item. The possible *improvement* step summarizes all procedures required to enhance the reliability, maintainability, or safety of an item, without a change to its original function. The method proposed in this thesis supports all four aspects of maintenance. [Deu10, pp. 26–28]

In the following sections, the terms *maintenance types* (Section 2.1.2), *maintenance strategies* (Section 2.1.3), and *maintenance management* are introduced. Whereas the types of maintenance can be divided with respect to the point in time, when an item is maintained [Deu12], the "maintenance strategy summarizes all management methods used in order to achieve the maintenance objectives" [Deu10, pp. 6–7]. Maintenance management connects the planning, control, and the improvement of activities and economics to fulfill the maintenance objectives defined above. Some of these objectives are presented in the following section.

2.1.1 Maintenance objectives

According to [Deu10], maintenance objectives are in general the assigned and accepted targets for maintenance actions. These targets could be:

- Availability
- Cost reduction
- Product quality
- Environment protection
- Safety and security
- Conservation of the item's value

[Str12, p. 4] defines the main task of industrial maintenance as the achievement of the preferably nondestructive useful life of a maintained object and to keep the required effort as low as possible. The dichotomy of both aspects leads to an optimization problem with respect to the associated maintenance costs, since a component's failure can be surely prevented only by a large number of maintenance actions.

In Figure 2.2, one example of the dichotomy of nondestructive useful life on the one hand and required effort on the other is presented by *measure costs* and *breakdown costs*, respectively. Measure costs include all costs related to the four aspects of maintenance (see Figure 2.1), e.g. wages of the maintenance personnel or costs due to troubleshooting or logistics. The breakdown costs include all consequential

costs due to downtime of the component (e.g. flight delays in commercial aviation). These costs are plotted against the degree (or number) of preventive maintenance actions. When the number of actions increases, the probability (and therefore the costs involved) of component failure decreases. At the same time, the costs due to maintenance measures rise exponentially, so that a total of both cost drivers results in a function with an optimum. Thus, the main objective is to apply a maintenance type near to or at the total cost optimum.

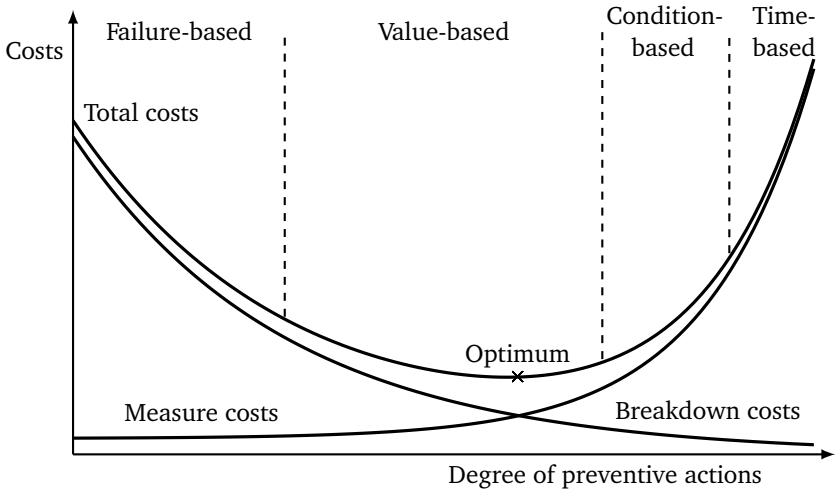


Figure 2.2.: Optimization problem of maintenance. Based on [Lei14, pp. 24–25]

Based on this figure, different maintenance types can be derived, which will be the focus of the next section.

2.1.2 Maintenance types

Since the definitions of different maintenance types are not consistent, some common approaches to distinguish them are presented in this section.

As depicted in Figure 2.2, [Lei14] distinguishes between four maintenance types with respect to their degree of preventive actions. The *failure-based* maintenance on the left of the plot marks the case when no components of an overall system are maintained until breakdown. On the right at a high degree of preventive actions, the two maintenance types *condition-based* and *time-based* are located. The high number of actions leads to a decrease in breakdown costs, but consequently to

higher measure costs. All three types are explained in more detail in Sections 2.1.2 to 2.1.2. Valued-based maintenance, which is a combination of all three above-mentioned types, represents the optimum of total costs; it demands the optimal maintenance type for every particular component of the overall system with regard to effort and availability.

Figure 2.3 presents another approach developed by [Deu10] to classify the different maintenance types. It divides the typical maintenance types into *corrective* and *preventive*. Here, similarities to the approach by [Lei14] are given, since corrective maintenance corresponds to the failure-based concept; condition-based and time-based (or predetermined) maintenance are both summarized under preventive maintenance. One difference is the further distinction between condition-based and predictive maintenance (PM), due to the application of prognosis tools to forecast the *remaining useful life* (RUL) of a component. This maintenance type is the focus of this thesis.

In the following sections, the most common maintenance types are introduced in more detail. It should be mentioned that there are several other maintenance types (e.g. risk-based [Mat10, pp. 133–147] or reliability centered maintenance [U.S10, section 5.5]) and more finely graduated distinctions such as predictive diagnostics or trending [VG15, slide 10], which are not within the scope of this thesis.

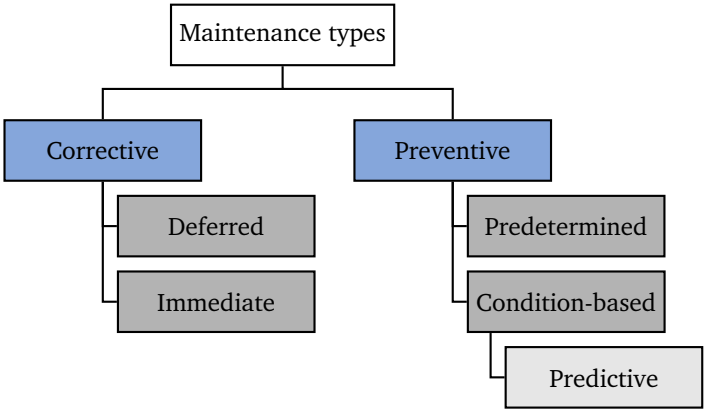


Figure 2.3.: Overview of the different maintenance types. Based on [Deu10, pp. 22–24]

Corrective maintenance

In accordance with [Deu10], corrective maintenance (also known as reactive maintenance [U.S10, section 5.2]) is carried out after a fault is recognized in order to return the component to a state in which it can fulfill its required function. This can be seen qualitatively for one particular component in Figure 2.4, where the reserve of wear-out is plotted over time with different maintenance approaches. Here, [Deu10] distinguishes between *failure* and *fault*. Whereas failure is an *event*, when the ability of an item to fulfill a required function is terminated, the fault describes a *state*, which characterizes the inability to fulfill the required function. Thus, a fault results from a failure. During its life cycle, a component can pass through four different states [Deu12]:

- *New state*: the requirements are completely fulfilled and may lie above the predefined limits for nominal state
- *Nominal state*: the item completely fulfills the requirements
- *Degraded state*: the fulfillment of the required functions is reduced, but within defined limits; it is located between nominal and minimum state
- *Minimum state*: the fulfillment of the requirements is only just ensured; a further degradation will cause a fault

The specification of the nominal and minimum state must be determined during the design phase of the component. In dependency of the application or safety specifications, the minimum state can also be set to the point, where the reserve of wear-out is completely exhausted. In parallel to the term *degraded state*, also the terms *latent* [Deu10, p. 19] or *incipient* fault [VLR$^+$06, p. xxi (prologue)] are also used to identify the deviation from *nominal state*.

In the case of the example degradation progress of Figure 2.4, corrective maintenance action is carried out in the first phase. Thus, the minimum state is crossed and the component is used until the point of complete exhaustion of reserve of wear-out. Due to the subsequent maintenance step, the component is (idealized) again in its nominal state or in case of an improvement, the nominal state can even be exceeded.

[Deu10] further classifies the corrective maintenance further into *deferred* and *immediate* corrective maintenance. The application of deferred maintenance is only feasible for components which are not relevant to safety or do not prevent the availability of the total system. The decision to choose one of these maintenance types leads to a trade-off between on the one hand the possibility of a total breakdown

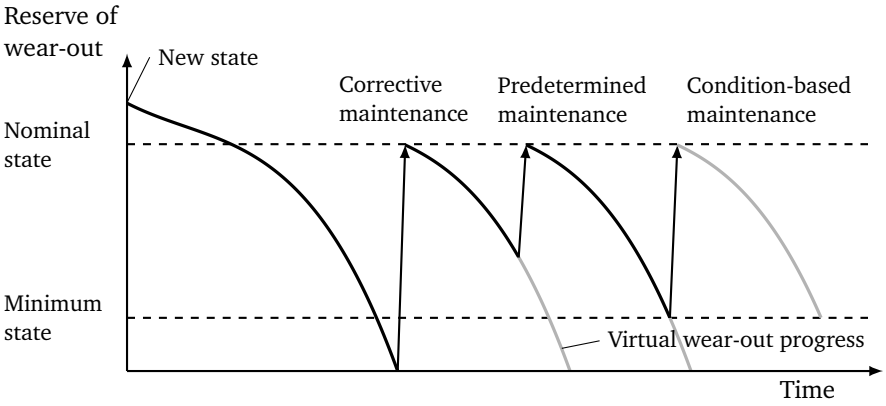


Figure 2.4.: Distinction between corrective, predetermined and condition-based maintenance. Based on [Deu12], [Str12, p. 300]

and on the other hand savings in maintenance actions. Thus, there is a critical point when immediate maintenance is more beneficial. [Bau08, pp. 102–103]

In comparison to this approach, the following three sections introduce maintenance types which ideally do not exhaust the complete reserve of wear-out of an item. Here, maintenance time is determined based on the time span between two maintenance actions (predetermined maintenance), the current state of the item (condition-based maintenance), or the forecast of the future wear-out progress (predictive maintenance). The virtual wear-out progress shown in Figure 2.4 illustrates the further degradation without any maintenance events, which would again lead to a total failure of the component.

Predetermined maintenance

The predetermined maintenance action is triggered according to established intervals of time, or the number of units of use [Deu10, p. 22]. For the calculation of the time between two maintenance actions, the wear-out progress is taken into account in order to prevent a breakdown before the component is exchanged. This therefore results in an unused RUL of a component, so that according to estimations of [Lei14, p. 16–17], about 85 % of predetermined maintenance actions are too early in the component's life cycle, as depicted in Figure 2.4. In this case, the component is exchanged before the minimum state is reached.

The benefit of this approach is the increased reliability of items in comparison to the corrective maintenance. In addition, downtime can decrease since the troubleshooting process can be omitted. From an economic point of view, the reduction of capital commitment due to spare part logistics is advantageous. However, these benefits are based on the assumption of an optimized predetermined maintenance. If there is a breakdown before the expected time of failure, the reduced storage of spare parts in particular will lead to increased downtimes. In addition, the data acquisition and data updating of the maintained components required to identify a proper maintenance interval are associated with high effort. [Str12, p. 298]

Condition-based maintenance

According to [Deu10, p. 22], condition-based maintenance (CBM) is a combination of condition monitoring (CM), inspection, testing, and analysis of the component. These investigations are triggered with respect to a schedule, on demand or continuously. The term CM covers a wide range of techniques which have been developed over the last 50 years to assess the current system or component condition. For example, these techniques examine the vibration signal or acoustic emissions of a rotating machine or its oil debris in order to distinguish between healthy and faulty components. Different signal types and common signal processing methods are presented in more detail in Section 2.3. [TL14]

There are several methods that can be used to define and structure the elements of CBM. Using the standards of [Int03], these elements are divided into data acquisition, data manipulation, state detection, health assessment, prognostic assessment, and advisory generation. Many of these elements are explained in more detail in Section 2.3. The industry association MIMOSA established one implementation and a suggested definition of an architecture for this standard. The implementation is called Open System Architecture for Condition-Based Maintenance (OSA-CBM) [MIM01], and its aim is to create a modular structure for CBM systems so that single components (e.g. for data manipulation) can be easily exchanged.

The advantages of CBM are similar to the benefits of predetermined maintenance. Additionally, as depicted in Figure 2.4 the reserve of wear-out can be exploited until the minimum state is reached; thus, the functionality of the component is always guaranteed. In contrast, due to high investment costs for measurement instrumentation or costs related to data acquisition and processing, CBM is not affordable for every application. Additionally, the mounted sensors can be weak spots, since a damaged or wrongly calibrated sensor leads to corrupted information about the current condition of the component. [Str12, p. 299]

Predictive maintenance

The final maintenance type is the predictive maintenance (PM), which is in the focus of this thesis. Based on the CM of CBM (as depicted in Figure 2.3), maintenance tasks are carried out as a result of forecasts that describe the future degradation process of the item, determined by significant parameters [Deu10, p. 23]. It should be pointed out that in the definition of CBM of [Int03] or [Bee04, p. 4], prognostic assessment is already a part of the CBM concept, whereas the distinction between both maintenance types proposed by [Deu10] is preferred in this thesis.

The core of PM is the forecasting method, i.e. the prognosis algorithm. The methods are divided into model-based, data-based, hybrid, and statistical or probability-based. New methods in the field of prognosis algorithms stem mainly from the first two approaches, so that different concepts in these fields are explained in more detail in Section 2.3.4. As motivated in Chapter 1, the predictions support the maintenance management related to the predictability of the component's failure, and are therefore able to further increase component life cycle, worker safety, and energy savings. Drawbacks arise from the still not verifiable statements about possible cost savings with the introduction of PM. In [Mob02, pp. 60–73], a survey in the field of manufacturing and production plants with regard to benefits from PM was executed. About 90 % of the participants reported improvements in maintenance costs and downtime [Mob02, pp. 64–65]. On average, the derived return on investment was 1.13:1. However, the actual savings varied among participants. Whereas 50.8 % did not recover any costs after the introduction of PM, 10 % achieved a return on investment of about 5:1. Since the actual affect of PM on costs is still a research issue, an assessment of possible savings with the application of the proposed PHM algorithm is presented at the end of Chapter 5.

2.1.3 Maintenance strategies

As previously mentioned, [Deu10, p. 7] distinguish between maintenance types and the maintenance strategy that determines how these types are combined in the context of maintaining complete systems. According to [Mik15, p. 16], there are three different strategies: *run to failure*, *on condition*, and *condition-based*. Whereas the run to failure strategy consists of corrective maintenance only, the on condition approach selects corrective maintenance for components that are not relevant for safety and security issues, and predetermined maintenance for all other components. The condition-based strategy permits a third option involving the predictions of PM. Thus, the maintenance intervals are selected dynamically.

The decision regarding which of the three strategies is to be selected in order to maintain a component is based on, for example, *failure modes, effects, and criticality analysis* (FMECA), which identifies the frequency of fault occurrence, the cause combination, and the consequences of faults. All assess the criticality of a component's failure [Mob02, p. 55]. A good introduction to the field of weak point analysis is given in [Str12, p. 159].

2.2 Integration of PHM into the field of maintenance

To support the predictive maintenance approach concerning decision-making, PHM systems are necessary. Thus, in this section the basics of PHM systems are introduced. Firstly, several definitions of important technical terms in this area are provided in Section 2.2.1. The requirements and several application fields of industrial PHM approaches are presented in Section 2.2.2.

2.2.1 Definitions and terminology

Although the disciplines of CBM, PM, and PHM have come up over the recent decades as significant technologies [VLR+06, Preface], the distinction between these terms is still not consistent. According to [TL14], the main difference between CBM and PHM is that CBM focuses on the monitoring of systems, whereas PHM is more concerned about life cycle management and the predictability of failures. In addition, the terms PHM and PM are often used as synonyms [VLR+06, p. 15]. As described in [LSGB15], the difference between these terms is that PHM assesses the reliability and forecasts the RUL of a critical system, whereas the decision to maintain the component based on this information is part of PM.

PHM in general "is a field of research and application which aims at making use of past, present, and future information on the environmental, operational, and usage conditions of an equipment in order to detect its degradation, diagnose its faults, predict and pro-actively manage its failures" [Kad13, p. 333]. Therefore, it combines interdisciplinary trends from the engineering sciences, reliability engineering, computer science, and others [VLR+06, Preface].

[VLR+06, pp. 1–2] describe the beginning of automated health management (i.e. fault diagnosis and prognosis of critical systems) in the application field of aircraft systems as follows: when systems grew in complexity, the manual detection of faults by means of equipment such as a voltmeter became impractical. Thus, as computer systems were introduced in the 1970s, manual detection was enhanced by the integration of built-in-test equipment (BITE) in line-replaceable

units (LRUs). In this approach depending on the observed component, test routines check the functionality of the LRU during the power-up process or the initialization step (so-called interruptive BIT) or continuously during normal operations (so-called continuous BIT) [PDN+01]. As the number of LRUs increased, mostly using different techniques to display the faults, there was again a trend to replace certain BITEs with sensors in order to directly sample and collect important signals for fault detection in global CM systems. Nowadays, CM systems also include the information of BITE on a higher hierarchy level for the overall assessment of the current health of the total system. [VLR+06, pp. 1–2, p. 177]

As shown in Figure 2.1.2, one advantage of extending CBM with a PHM approach is the introduction of failure forecasting methods based on the condition of the component. The entire process of PHM systems from data collection to the forecast is depicted in Figure 2.5. This process is roughly divided into *signal generation and signal processing*, *diagnostics*, *prognostics*, and *advisory generation*. In the first step, the data is collected. In accordance with [HW14, p. 1], the subsequent steps, consisting of *data preprocessing*, *feature generation*, and *feature reduction*, are parts of signal processing (or data manipulation as defined by [Int03]), the goal of which is to extract the important information of a signal. During data preprocessing, the signal to noise ratio is increased by means of filter methods (low-pass, notch etc.) for example. The feature generation includes signal compression and certain transformations (Fourier transformation, Wavelet transformation) depending on the examined signal type [VLR+06, pp. 97–104]. Due to the large volume of data, especially after the transformation into frequency domain, certain features that accurately determine the current state of the item must be found. Therefore, the feature reduction step is applied. By means of these features, a fault classification is executed in the context of fault diagnostics. [Ver99] and [VLR+06, p. 176] link fault diagnosis to fault detection, fault isolation, and fault identification techniques, so that the fault is recognized, localized, and quantified. According to [Int03], the output of the diagnostic part is – amongst other things – a *health indicator*, which ranges from 0 (complete failure) to 10 (new state). Based on this health indicator and the estimated fault type, the future health state and failure mode of the component is determined in terms of a prognosis part. Here, [Int03] also take into account the projected loads on the item in order to increase the accuracy of RUL estimations. The consideration of load is also applied in this thesis. [VLR+06, p. 284] describe the major challenge in prognosis as the management of several different uncertainty types such as future fault evolution, future load, or the diagnosis uncertainty. Thus, the output of the prognosis is for instance a probability density function (PDF) in the case of continuous RULs or a probability mass function (PMF) for discrete RULs. Based on the PDF, an *advisory generation* about

necessary maintenance actions is published and transferred to the maintenance personnel.

In the context of PM, a decision making step with regard to possible maintenance tasks based on the results of the advisory generation step would follow. Several approaches for each particular step are presented in Section 2.3. The terms diagnosis and prognosis, which were briefly introduced in this section, are specified in more detail in Section 2.3.3 and Section 2.3.4, respectively.

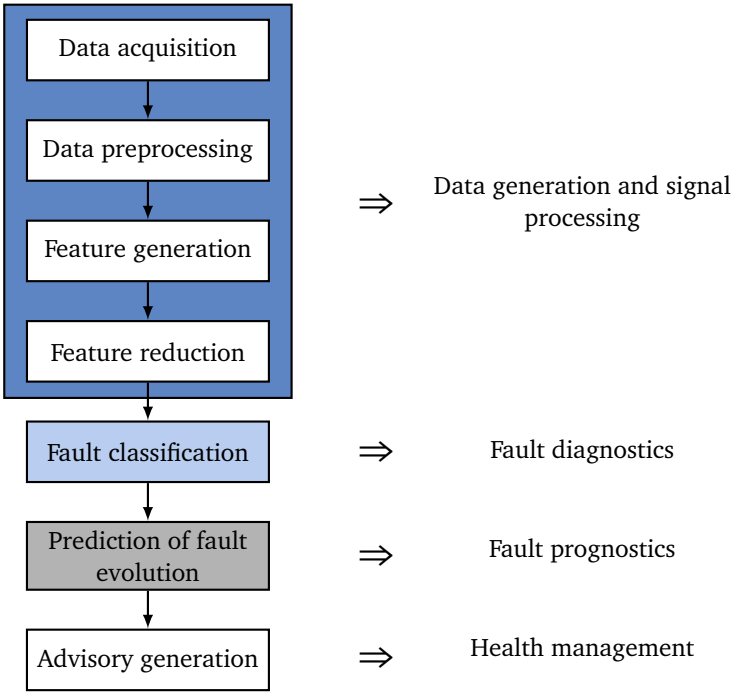


Figure 2.5.: Basics steps of a PHM system. Based on [VLR+06, p. 15] and [Int03]

2.2.2 Requirements and application fields

The critical step related to the introduction of PHM systems is the interaction of sensor accuracy, diagnostic, and prognostic effectiveness. The accuracy of the predicted fault evolution can ensure a support of the maintenance management only, if a fault can be detected and classified in early stages. Furthermore, there must be a

time span between fault detection and minimum state, i.e. an evolving degradation. Not every system matches these requirements. [VLR+06, p. 4]

One desirable characteristic of degradation is also monotonous behavior. However, due to factors such as recovery effects this characteristic is not always obtained in real applications. Thus, the PHM algorithm must be able to accurately predict the RUL of components, even with non-monotonic attributes. [Jen13, p. 51]

The application domains of PHM systems are widespread. Beside the mechanical, thermal, and electromechanical industry, the electrical and electronic sectors have recently applied PHM approaches to their systems [VLR+06, p. 7]. [VP06] describe methods for failure prediction of electronic power systems by observing current drain or temperature. [SCWG09] apply a prognostic approach to determine the RUL of transistors in the area of semiconductors by measuring the collector-emitter current, which changes over time due to degradation. Another typical electro-chemical application is the degradation of accumulators such as lithium-ion batteries ([BSGR14] or [SGC09]) due to their characteristic, exponential fault evolution, which is measured by changes in capacity. However, the assessment of electronic systems in terms of health management is still not as well established as it is for mechanical or in particular electromechanical systems [VP06]. The sectors of industry involved are manifold; automotive, aviation, military, or in general, the manufacturing sector have been the main key enabler for PHM systems. However, the windmill, unmanned vehicles, or the robotic industries also introduced PHM in their systems [PBG12]. Due to its benefits with regard to safety and overall costs, the aviation industry in particular is interested in PHM systems [VLR+06, Prologue]. Besides the structural health monitoring of aircraft structures such as airframe, wings, landing gear etc. (see [Jen13, pp. 178–193] for more information), jet engines in particular are equipped with an immense number of CM systems to detect faults in their early stages and to transfer this information via satellite to support centers [Cal09]. Thus, PHM is a key element to justify the choice of single-engine aircraft [MK07].

On a component level, only a few applications are applied by researchers for the development of PHM systems and algorithms. Beside the previously mentioned batteries, components of rotary machines such as gears ([DQQG15], [WMZZ15]), shafts ([JMC+09]), or bearings in particular are at the center of research. According to [Ant07] and [BBV14], bearings in particular offer many benefits, such as

- Wide range of application
- Low price per unit
- Prone to faults
- Continuous, stochastic degradation

- Extensive experience in CM
- Characteristic fault signature.

These components are therefore interesting for the generation of degradation data.

[Pow85a] and [Pow85b] initiated a survey to investigate the reliability of large induction machines such as asynchronous motors, DC motors etc. greater than 200 hp (≈147 kW) and especially to determine common failure causes under real conditions in the daily business of several industry sectors. In summary, the survey showed that bearing faults were the cause of about 41 % of 360 total failures over several years [Pow85a]. Further readings concerning the number and kind of maintenance actions [Pow85a] or the analysis of ambient conditions or failure contributors [Pow85b] for example are recommended. Consequentially, the combination of asynchronous motor and bearing is the exploratory focus of many publications due to the widespread use of this robust type of electric motor [SLNL14] and the extensive experience of CM in terms of bearings. Thus, this combination is also the center of a case study in Chapter 4 and Chapter 5.

2.3 Examples of approaches to basic steps of PHM algorithms

The following sections present a survey regarding approaches that are state of the art for the basic PHM steps presented in Figure 2.5. Since an overview of all possible application fields would go beyond the scope of this thesis, the focus is narrowed down to the applications and components that are the main focus of PHM research publications especially in the area of rotary machines such as gear boxes, bearings, and other applications. Thus, just a small set of approaches is provided with no guarantee for completeness. The introduced methods are presented with exemplary publications, in case a detailed look is desired.

2.3.1 Typical signal types

Although the types of signal which are tracked within the CM concept always depend on the application, several signal types are indispensable for the health monitoring of a wide range of cases. [Ver99] provide a detailed overview of measured signals for the condition detection of a variety of applications. They subdivide the signal types into mechanical quantities (e.g. pressure, torque, etc.), electrical quantities (current or power input), vibration and noise (sound level, amplitudes etc.), and many others. A short extract including the most relevant types for this thesis and their related applications is shown in Table 2.1.

One of the most important signal sources, especially in case of rotary machines, is vibration. In context of CM, vibration always corresponds to structure-borne

Table 2.1.: Typical signal types in CM and exemplary use cases

Signal type	Use cases
Vibration	Gear box (spline section of helicopter) [OCH+14], pumps [LDP12], bearing [MS85]
Temperature	Transistors [SCWG09], heat exchanger [WNS+14]
Current	Battery (electric vehicle) [BSGR14], transistors [SCWG09], bearings [Han10], gear box [MK06]
Torque	Shaft [BCFP14]
Position	Bearing (of tidal turbines) [GCLR14], blade tip [TvFM01]
Acoustic emission	Blades (combustion engine) [WC05], bearing [EM10]
Debris	Gas path (F-35 propulsion system) [PN06], gear box (gears and bearings)[Dup10]

sound, whereas air-borne noise is known as acoustic emission. Another approach used to distinguish between these is the frequency range: [Hun96, p. 259] defines the range for vibration from 1 Hz to 25 kHz for structure-borne sound and from 100 kHz to 1 MHz for the acoustic emission. Benefits of vibration analysis are

- Good representation of the current machine performance [LWZ+14]
- Most effective method to detect many different types of mechanical faults [GS04, p. 7]
- Wide range of research effort

Thus, new approaches to the diagnosis and prognosis of bearing faults for example are mainly evaluated based on the analysis of vibration signals.

Another example of frequently investigated signal type in terms of rotary machines is the motor current of asynchronous motors. Here, mechanical faults such as the wear of gears and bearings or electrical faults such as broken rotor bars of the motor are captured by the analysis of the phases of the motor current at fault specific frequencies [MK06]. According to [Kli90], the detection, isolation and identification of bearing faults by means of motor current signature analysis (MCSA) in particular has increased due to a greater appreciation of the benefits of CM and the wide availability of low-cost personal computers. Although the diagnosis based on MCSA is wide spread, no PHM system has previously been evaluated

with it. Thus, the MCSA in terms of bearing faults is introduced in more detail in Section 4.1.1.

2.3.2 Signal processing in PHM

After the data is collected, the signals must be adjusted for the diagnosis and prognosis steps. As depicted in Figure 2.5, signal processing in PHM is divided into data preprocessing, feature generation, and feature reduction so that the output is a health indicator, which is analyzed in terms of the subsequent diagnosis.

2.3.2.1 Data preprocessing

The goal of the preprocessing step is to remove artifacts and to reduce the noise and quantity of data [VLR+06, p. 96]. Thus, in general this step consists of filtering and a normalization step. The applied filters are conventional digital Butterworth filters such as low-pass filters (to reduce noise in high frequencies [VLR+06, p. 98]), notch filters (e.g. to remove outshining frequencies), or band-pass filters (e.g. when the frequency range of characteristic fault features is known [AM01]) [Alp10, p. 138]. An adequate alternative for Butterworth filters are Wavelet filters, which can be designed in the same way. Due to the availability of more options with the Wavelet transformation in comparison to the Fourier transformation, the application of these filters can be beneficial in certain use cases [VLR+06, p. 126].

Since rotary machines are the focus of most PHM approaches, the normalization of the resulting oscillating signals is a common step. Two different approaches are introduced in [Agg15, p. 37]: *min-max scaling* and *standardization*. Both are statistical methods to reduce the variance of the signal. Whereas the output of a min-max scaling is a representation of the signal within the range of 0 to 1, in the case of the standardization method, the signal's mean value is set to 0 and its standard deviation is 1. Thus, if the signal is normal distributed, the range of the signal after standardization is between -3 to 3. [Agg15, p. 37]

2.3.2.2 Feature generation

[Agg15, p. 28] describes the feature generation step as transferring low-level signals to high-level features. These features are either directly passed on to the diagnosis or are further reduced in the context of a feature reduction step, which is introduced in the next section. It must be mentioned that the term *feature generation* is widely known as *feature extraction*, whereas in this thesis, the feature extraction is a sub-item of the field of feature reduction.

Figure 2.6 gives an overview of the typical feature generation methods summarized by [YQI08] and [DGO11]. Comparable to the choice of measured signal type, the selection of an appropriate method is also application-dependent. Thus, the methods are subdivided into the cases of stationary and non-stationary signals.

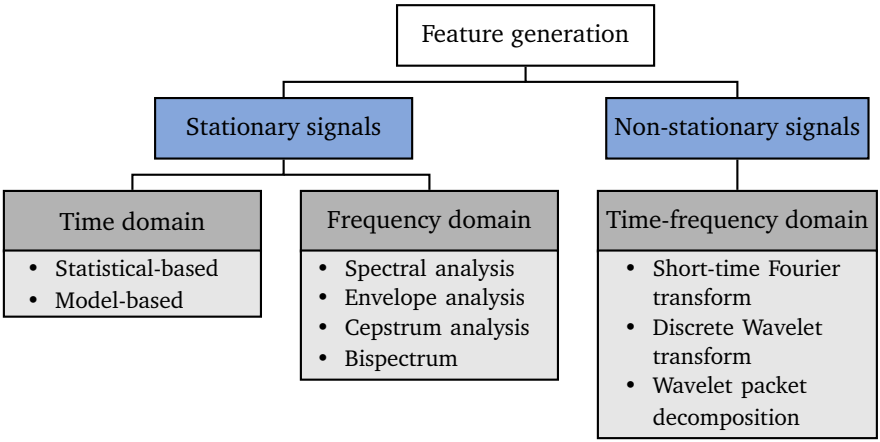


Figure 2.6.: Overview of the different feature generation methods. Based on [YQI08] and [DGO11]

Stationary signals are further classified according to the domain in which the features are generated. The time domain procedures include statistical and model-based methods, whereas in the frequency domain all methods are based on the Fast Fourier Transform (FFT). The theory of FFT can be found in relevant literature such as [HW14, pp. 128–134].

[YQI08] describe the time domain features as being suitable for fault detection in the context of the subsequent diagnosis. Since one requirement of feature operations can be real-time capability [VLR+06, p. 104], the use of traditional statistical-based approaches in particular is widespread. These include quantities to determine the power content of the signal such as root mean square (RMS), or values to find patterns in the signal such as the peak-to-peak intervals or crest factor. Classical features also include the four statistical moments (mean, variance, skewness, and kurtosis). All methods are significant, since they can be calculated directly on raw signals. [YQI08]

One important element of model-based feature generation methods is the modeling of the examined system, for example by means of differential equations which

represent the physical behavior of the system. One approach to gathering features is to estimate the parameters of these models based on the measured time series and to use these parameters as features (see [DSG12] for the modeling of pumps). Another approach is to model the nominal state of the tested unit. The residua between the output signal of this model and the measured signal then serve as features (see [JOL08] for the application of bearings). A deeper introduction to model-based approaches in the field of PHM is presented in Section 2.3.3.1 and Section 2.3.4.1.

Whereas time domain features are mostly appropriate for fault detection, the frequency domain methods are characterized by additionally localizing the faulty component of a system during the diagnosis step [YQI08]. Since most CM signals of rotary machines are oscillatory (vibration or motor current signals), transformation into the frequency domain is appropriate. Here, the most popular approach is spectral analysis of the FFT or the power spectral density (PSD) of the signal. The PSD is the FFT of the autocorrelation function of the signal [HW14, p. 200]. These amplitudes are either evaluated directly or further processed, for example by calculating the statistical moments of these features (see [PAK14] in the context of bearing faults). One similarity of the other three approaches in the frequency domain is that either the signal or spectrum itself is further processed. The envelope analysis consists of band-pass filtering in the area of interesting frequencies, and enveloping to remove structural resonances for example (see [RAC01] for cyclostationary machines). The aim of cepstrum analysis is the detection of the periodicity of the spectra. This is achieved by means of the power spectrum of the logarithm of the power spectrum [YQI08]. It is expected that a fault will be expressed by impulses in the vibration signal for example. Cepstrum analysis separates this convoluted signal, which consists of these impulses and the impulse responses of the system (see [RS16] for gears) [HW14, p. 340]. The final frequency domain method is the bispectrum, which is simply the FFT of the second statistical moment of the underlying signal. One benefit is a reduction in Gaussian noise, so that the fault frequencies are amplified (see [Han10] for bearings). The examination of higher order spectrum (trispectrum etc.) is also common.

The third of these methods is recommended if the examined signals are non-stationary e.g. due to fast load changes or engine startups or shutdowns, respectively. Thus, the sensor signal is transformed into the time-frequency domain in order to identify time-dependent variations in the spectrum [YQI08]. Here, the short time Fourier transform (STFT) is one classic method, which is based on the conventional FFT. The STFT is calculated by transforming a sliding window of the signal into the frequency domain, so that the so-called spectrogram includes temporal and frequency information. The challenging aspect of this method is the

selection of the window function (Hamming, von-Hann, etc. see [HW14, pp. 186–187]) and an appropriate window length (see [Nad04] for bearings). The Wavelet transformation can be explained analogous to the STFT. The abovementioned window function is replaced by a wavelet (Morlet, Haar, Daubechies), which is slid over the CM signal [HW14, pp. 294–296]. The advantages of wavelets are that they are expressed by a dynamical length (realized by a dilation parameter) instead of the constant length in the case of the window function of the STFT. Hence, higher frequencies are analyzed with high frequency wavelets, i.e. shorter window length, and vice versa so that particular frequencies are represented more accurately [HW14, p. 286]. One extension of the Wavelet transformation is the Wavelet packet transform, the goal of which is to further increase the resolution in higher frequencies. A detailed introduction to the Wavelet packet transformation and a case study with bearings is presented in [ED04]. The crucial aspect of both Wavelet transformations is the selection of a suitable mother wavelet. References for a structured choice are presented in [HW14, pp. 303–307].

2.3.2.3 Feature reduction

The final part of the signal processing in terms of PHM systems is feature reduction. The aim of this step is to identify these features and summarize them into a fault feature vector, which allows an accurate distinction between the different damage states of the observed component [VLR+06, p. 106]. According to [Alp10, pp. 109–110], there are several reasons for feature reduction, i.e. reducing the data dimensionality:

- Memory and computation requirements are reduced
- Simpler models based on less data are more robust
- Models have less variance due to fewer outliers
- More knowledge is extracted from fewer features
- Benefits the results of visualization and analysis

The main problem with a large feature dimension is the so-called *curse of dimensionality*, which is outlined in [Bis13, pp. 33–38] for example. This says that by increasing the input dimension, the amount of training data must rise exponentially in order to cover all possible necessary feature combinations. Since in most applications the amount of training data is limited, a suitable feature dimension must be determined in upstream analyses.

The feature reduction step is divided into *feature selection* and *feature extraction*. The goal of feature selection is to find a smaller feature dimension k out of n, i.e. a subset that represents the most information so that $(n - k)$ features are discarded. Thus, to attain this subset, one approach is to filter and rank features based on metrics of the information theory [CP04]. There are several state-of-the-art metrics in the field of statistics that can be used to represent the information density of signals. Besides simple signal variance, the signal's entropy in particular is a metric which is at the center of information theory. To rank two different features, relative entropy and mutual information (see Section 3.2.2 for a detailed introduction) are state-of-the-art methods. A small introduction into the information theory is presented in [Bis13, pp. 48–58].

To calculate the feature subset based on the information density, there are two widely used methods: sequential forward selection and sequential backward selection. Sequential forward selection begins with no features and in each iteration accepts only features that increase the overall information density, until no further gain is achieved. In contrast, the sequential backward selection begins with all features and discards features analogously [Alp10, pp. 110–112].

Another approach to obtaining a feature subset in terms of CM is the identification of certain characteristic fault features. One typical example is the bearing fault frequencies proposed by Palmgren [Pal59]. Based on the geometry of the bearing, the revolution, and the contact angle, the equations of Palmgren provide evidence regarding the frequencies of several bearing faults so that other frequencies can be discarded.

On the opposite side in the case of feature extraction, a new set of k features is calculated by combining the n original dimensions [Alp10, p. 110]. The most widely used methods are Principle Component Analysis and Linear Discriminant Analysis. Both methods are introduced in [Alp10, pp. 113–133].

2.3.3 Fault diagnosis

The aim of the diagnosis step is the detection of anomalies due to faults that create deviations from the nominal system behavior [Jen13, p. 28]. This deviation is expressed with a health indicator, which represents the relation between the current state and the predefined threshold for the minimum state. It is a result of the three steps of fault detection, isolation, and identification, which are the basis for diagnosis and are summarized by [VLR+06, p. 176] as follows:

- Fault detection: Discovery and report of abnormal operating conditions
- Fault isolation: Determination of the failing component

- Fault identification: Estimation of the nature and extent of the fault evolution

With respect to the examined application, only one (fault detection), two (fault detection and isolation), or all three steps are necessary for the decision making of subsequent maintenance tasks. Since the PHM algorithms which are at the center of this thesis, are based on health indicators, all three steps are mandatory. Thereby, typically not only one fault type, but several faults are observed in parallel. In [SCR13], the electrical damage (broken rotor bar) and mechanical damage (bearings) of an induction machine are diagnosed. Another output of multi-fault or single-fault diagnosis is classification uncertainty. Since different faults have similar effects on features, considering the probability of each fault is necessary. In [JCD+15] several typical faults of bearings are diagnosed simultaneously.

There are two main approaches to fault classification: *model-based* and *data-based methods*. Comparable to the model-based feature generation in Section 2.3.2.2, model-based diagnosis approaches are also dependent on an analytical representation of the physics of the process. With reference to the approach, model-based diagnosis uses the same models as model-based feature generation. On the other hand, data-based methods rely on mostly statistical models using the basis of *historical process data*. The methods to form these statistical models are mainly adapted from *machine learning* and *pattern recognition approaches*. A deeper look into model-based and data-based fault diagnosis is provided in the following sections. [VLR+06, pp. 178–179]

2.3.3.1 Model-based diagnosis

Figure 2.7 provides an overview of a model-based diagnosis approach. Here, the actual process is modeled by differential equations or if-then rules for example, which form the process model. The input for both the model and the real system is the input $\boldsymbol{U}$. The output of the actual process is Y, which is also considered to be an input of the model. Based on the feature generation and reduction methods presented in the previous two subsections, a feature vector of residua $\boldsymbol{\epsilon}$ (difference of measured output $\boldsymbol{Y}$ and estimated output of the model), system parameters $\boldsymbol{\Theta}$, or state variables $\boldsymbol{x}$ is generated. By comparing this feature vector with the feature vector of the nominal behavior, the fault diagnosis determines a health indicator containing the fault type (fault isolation) and the fault progress (fault identification). [Ise06, p. 62]

[VRYK03] further differentiate the model-based diagnosis methods into qualitative and quantitative. Qualitative approaches are based on qualitative functions

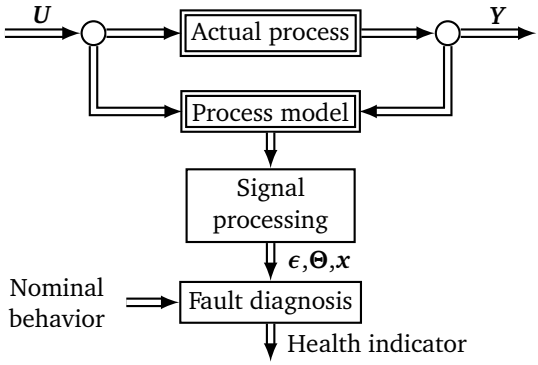


Figure 2.7.: Block diagram of model-based fault diagnosis. Based on [Ise06, p. 62]

(basically if-then-else rules) in cases where the exact model parameters are not known or the system's complexity is too high. Examples can be found in [VRK03]. More popular approaches are in the field of quantitative model-based methods such as parameter estimation by means of Kalman filters. By estimating different parameters, fault classification and quantification is determined [Sch11, p. 34].

2.3.3.2 Data-based diagnosis

[VLR+06, pp. 181–191] present a large variety of data-based fault diagnosis methods from simple approaches of fault detection such as alarm bounds to more complex systems such as artificial neural networks (ANNs) for fault identification. However, the similarity of all these methods is a model resulting from a training step which includes historical process data. By comparing Figure 2.8 with Figure 2.7, several differences between model-based and data-based diagnosis are derived: the output $\boldsymbol{Y}$ is directly transformed into a feature vector $\boldsymbol{\Phi}$ during a signal processing step. Subsequently, the feature vector is analyzed on-line with regard to anomalies by means of the trained model. One result of this fault diagnosis step is again a health indicator for specific faults.

[VRKY03] differentiate again between qualitative and quantitative data-based diagnosis methods. One qualitative approach is limit checking based on the current data in relation to predefined thresholds. The drawback of this method is that only large deviations of the nominal state are detected. Thus, transient fault states that are mandatory for PHM algorithms are not identified. [Sch11, pp. 28–29]

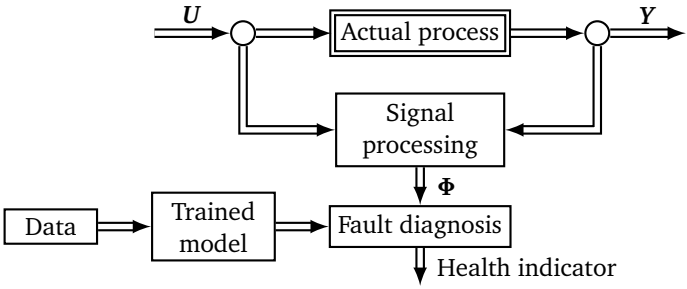


Figure 2.8.: Block diagram of data-based fault diagnosis. Based on [VLR+06, p. 179]

In contrast, state-of-the-art methods are mainly based on quantitative techniques of machine learning. Here, supervised and unsupervised learning algorithms are both applied for the diagnosis of faults. Whereas supervised algorithms are based on *labeled data* (i.e. the data is directly related to a fault state or nominal state) unsupervised algorithms work with data that are *unlabeled* [Jen13, pp. 30–32]. The latter is mainly applied to reduce the data dimensionality and can therefore also be applied for feature reduction.

Many approaches with varying complexity have been introduced in the field of unsupervised data-driven fault diagnosis. [SLH+95] and [MPP07] use ANNs to identify fault patterns in unlabeled data of induction machines. By summarizing similar fault occurrences in particular clusters, different fault types (bearings, shaft eccentricity, stator winding) are diagnosed. In [MPP07], the severity of the fault is also classified into *no fault*, *small fault*, and *severe fault*. [YGA11] apply K-means to diagnose different bearing faults. They summarize the advantages of the K-means algorithm as:

- Simple and widespread learning algorithm
- No prior knowledge about data necessary
- Linear time complexity in size of dataset
- Convergence is assured

One drawback is that the results are not deterministic and vary with respect to the initialization of the algorithm. The basics of K-means are introduced in detail in Section 3.3.1.

Supervised algorithms are applied to two tasks [Jen13, p. 36]:

- Classification: Groups input data into two or more classes (nominal–faulty state)
- Regression: Assigns objects or data to continuous labels (health indicator 1–10)

As a result, classification algorithms are typically used for fault isolation and identification based on the sampled feature vector. To divide the input data into several classes, so-called decision boundaries are defined in most common supervised classification algorithms [Bis13, p. 179]. One classical supervised classification method is the support vector machine (SVM). In [PPG+13], the SVM is used to identify different fault types in the case of a truck engine.

On the other hand, regression algorithms are applied to model a certain system behavior analogous to the approaches of Section 2.3.3.1. However, the models created by regression algorithms are non-parametric and, thus, no knowledge of the process physics is required [Sch11, p. 27]. In [KP02] an ANN is applied to rebuild the nominal state of an induction motor based on the motor current signal, revolution, and the voltage. When a mechanical or electrical fault occurs, the residuum of the data-based model and the measured signals is evaluated.

2.3.3.3 Performance metrics for fault diagnosis

Several metrics to evaluate the performance of the fault classification are well established. One example is the *confusion matrix*, which assesses the correlation between the predicted and actual fault condition. The principle of the confusion matrix is presented in Table 2.2. In the columns, the predicted classes (e.g. nominal and faulty) of the classification algorithm are presented, whereas in the rows the true classes are depicted. The options *positive* and *negative* correspond to the outcome of the predicted class related to the hypothesis "*Is the component faulty*". The terms *true* and *false* describe the correctness or miss of the predicted class. Thus, a classifier with entries only on the main diagonal always correctly classifies the current state of the component. True positives are also called *hits*, and false negatives are called *misses* or *missed alarms*. The other two possible classification terms are false positives (or *false alarms*) or true negative (or *correct rejections*). With regard to safety and costs, the misclassification cases show different consequences: a high false alarm rate (or *no failure found* (NFF) rate) leads to an increase in life cycle costs, since the RUL is not exploited. On the other hand, a high missed alarm rate can result in possible safety issues. [Jen13, pp. 41–42]

Other metrics such as the receiver operating characteristics (ROC) curves are introduced in [Jen13, pp. 42–44].

Table 2.2.: Confusion matrix of two classes. Based on [Alp10, p. 489]

		Predicted class	
		faulty	nominal
True class	faulty	true positive	false negative
	nominal	false positive	true negative

2.3.4 Fault prognosis

Once a degraded state is determined, the prognosis of the component's RUL is initiated. In Figure 2.9, one artificial example of an RUL prediction is presented in order to introduce several terms in the context of fault prognosis. The real degradation is plotted over time from new state to minimum state at time $t = 100\,d$. The degradation of a component is defined by [Deu10, p. 16] as a "detrimental change in physical condition" and is expressed by measuring related changes in signals [VLR$^+$06, p. 334]. One typical signal type that qualitatively represents the degradation of a rotary component is the RMS of the vibration signal. Compared to the reserve of wear out shown in Figure 2.4, degradation shows a reverse behavior.

From the new state ($t = 0\,\mathrm{d}$) to minimal state, two thresholds are crossed. One level is reached at $t = 22\,\mathrm{d} = t_p$, when the effect caused by the incipient fault is detectable by the diagnosis algorithm; until then, predictions are not possible [VLR$^+$06, p. 259]. The second threshold that represents the minimal state is represented by the *hazard zone*. In comparison to a particular failure threshold (as marked in Figure 2.4) a hazard zone is defined by its lower and upper bounds or can be expressed with any probability density function (PDF) of a given probability distribution [VLR$^+$06, p. 298]. A Gaussian distribution for the hazard zone is a common assumption.

One prediction at $t = 60\,d$ is plotted. As mentioned in Section 2.2.1 the RUL is predicted by a PDF generated by the prognosis algorithm. Thereby, the algorithm covers the uncertainty due to [Jen13, pp. 50–51]:

- Stochastic degradation process
- Differing load levels
- Measurement uncertainty
- Varying production tolerances
- Modeling uncertainty
- Noisy diagnosis output

Thus, an uncertainty bound that represents $\pm 3\sigma$ of this distribution and its mean is depicted. Assuming a Gaussian distribution in predicted degradation distribution as well as in the hazard zone, one possibility to determine the *predicted RUL* is depicted in Figure 2.9: the time difference between the point, when the mean of the degradation distribution crosses the mean of the hazard zone, and the starting time of the prediction represents the estimated RUL. In contrast, the *actual RUL* is the time difference between the start of the prediction and the time when the real degradation crosses the mean of the hazard zone. The deviation of predicted RUL and actual RUL serves for several prognosis performance metrics, which are introduced in Section 2.4.

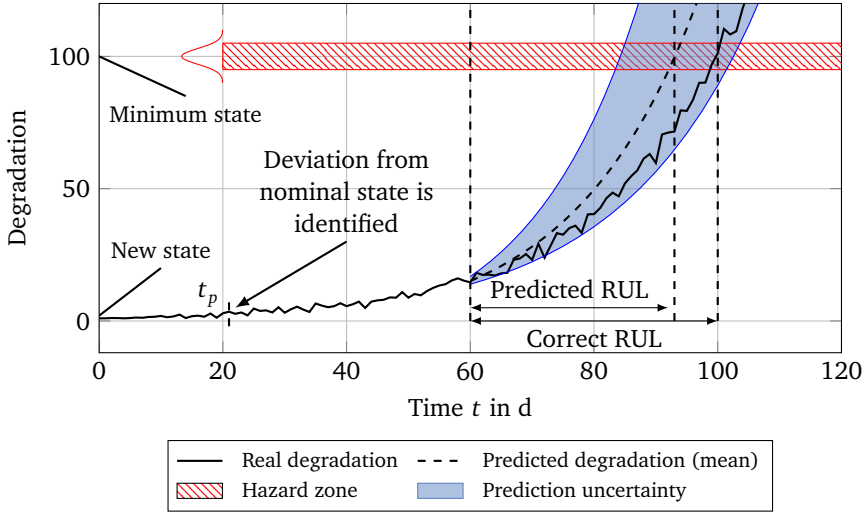


Figure 2.9.: Example of a prognosis at time $t = 60\,\mathrm{d}$

[Jen13, p. 52] distinguish between four kinds of approaches for prognosis algorithms: *model-based*, *data-based*, *hybrid*, and *statistical* or *probability-based* methods. Whereas hybrid approaches are a combination of model-based and data-based techniques, probability-based methods use the history of component failures and operational usage profiles. By fitting probability distributions such as Weibull to this data [SR03], predictions of maintenance intervals in the context of predetermined maintenance are derived. Both, hybrid and probability-based approaches are not in the scope of this thesis, so only the widely used model-based and data-based methods are introduced in the following sections.

2.3.4.1 Model-based prognosis

According to [Jen13, p. 51], the basis of a prognostic algorithm is a damage propagation model that captures the fault evolution in the future with reference to operational and environmental conditions. The prognosis algorithm applies this damage propagation model, combines it with uncertainties (current state uncertainty, model uncertainty etc.), and determines the prediction uncertainty.

Based on the dynamical model and the features derived from the diagnosis step, the prognosis in terms of model-based approaches is triggered. Here, the dynamical model is typically integrated in nonlinear filters such as particle filters or extensions of Kalman filters (Unscented Kalman filter, Extended Kalman filter) to capture modeling or measurement uncertainties. The dynamical model is then connected to a damage propagation model. This model stems mainly from a wear equation such as the Paris' law, which expresses the grow of cracks in mechanical parts [VLR+06, p. 292]. A prediction is achieved by iteratively propagating the future fault level (based on the damage propagation model) and the subsequent output of the modeled system, until the thresholds for minimum state are crossed. The uncertainties of the RUL prediction are determined as a result of the nonlinear filters. These steps of a model-based prediction are summarized in Figure 2.10 with the feature vector $\mathbf{\Phi}$ of the signal processing as input and the kth RUL probability distribution as output.

One model-based approach for both diagnosis and prognosis is presented in [DSG12]. The examined application is a simulated centrifugal pump. By capturing the physics of mechanical (bearings, rotating shaft), electro-mechanical (motor), and fluidic components (impeller), the dynamical behavior is modeled. These models are integrated into three different nonlinear filters. The two fault cases are impeller wear and bearing fault. Both are modeled with a particular wear equation for damage propagation. The results show nearly perfect RUL predictions for every filter, even at the beginning of the life cycle.

2.3.4.2 Data-based prognosis

Model-based prognostics can result in excellent prediction performances due to a high level of detail in modeling degradation process of the components. Hence, in [Jen13, p. 52], the application of model-based prognostics is assumed to be an enabler for prognostic algorithms in complex systems such as aircraft. However, if a dynamical model is not able to capture all physical processes during the degradation process or the modeling process is not profitable, data-driven methods are applied to obtain a damage propagation model. The variety of methods and ap-

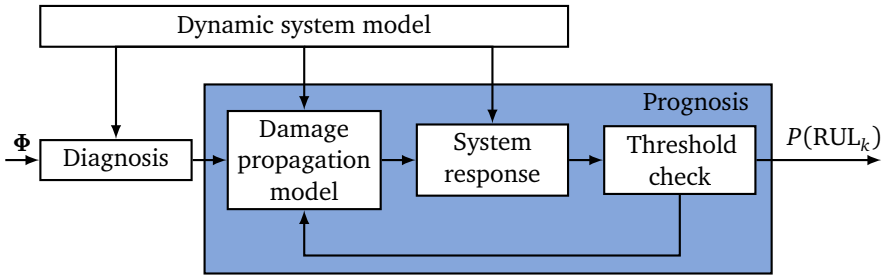


Figure 2.10.: Block diagram of model-based fault prognosis. Based on [DSG12]

plication fields in the area of data-driven approaches has increased greatly in the recent years due to progress in the field of machine learning. Certain benefits of data-driven approaches stem from self-adapting algorithms, as well as less a priori information about the examined process, and thus, advantages in cost-benefit analysis in comparison to model-based approaches. One requirement is the provision of sufficient test and training data that describe the damage progress. If degradation data is not available, the artificial generation of realistic damage can be time-consuming and detrimental for the cost-benefit. [Jen13, pp. 55–56]

One approach to data-based prognosis is the selection of an analytical model (n-order polynomial, exponential function etc.), which is included in filters. The online adaption of the inherent parameters is based on sampled data. This approach is known as model identification and is sometimes allocated to model-based approaches. One example of this method is presented by [KAS12] to predict the RUL of spindle tools. The spindle power is monitored and a 2nd order polynomial is used as a damage propagation model. By including a random sample path method, the RUL distribution is estimated.

Other approaches use methods from the field of machine learning and data mining [KBA08]. State-of-the-art techniques in the context of fault prognosis are regression methods. As mentioned in Section 2.3.3.2, regression models predict an unknown function value t_{n+1} at input value x_{n+1} based on a training input vector $\boldsymbol{X} = (x_1 ... x_n)^T$ and a training output vector $\boldsymbol{T} = (t_1 ... t_n)^T$ [Bis13, pp. 137–138]. One example of how to generate a regression model based on training data points, is shown in Figure 2.11. Several methods to create a regression model are introduced in [Bis13]. These vary from simple linear regression based on linear basis functions to more complex methods in the field of *kernel methods* or ANNs. Further reading into this field is recommended in [Bis13, pp. 291–356].

One data-driven approach to developing a feasible fault propagation model is to train a regression model f with the measured degradation d_k (or operational

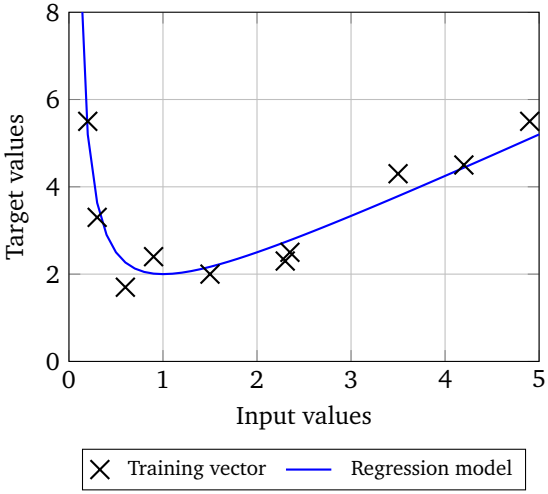


Figure 2.11.: Example of a regression model

conditions) as input data and degradation rate $\dot{d}_k$ as outputs, also known as *targets*. The RUL is then calculated by iteratively determining the new degradation $d_{k+1} = d_k + f(d_k)$, until a threshold is crossed. The RUL uncertainty is generated by including this propagation model into a filter approach (e.g. particle filter), which includes model and measurement uncertainties, similarly to the model-based approach. This extension also allows the use of several propagation models in parallel to forecasting different types of degradation progresses. An application of this approach is presented in [KBA08].

Another approach to forecasting the RUL is the assumption that a component runs through certain discrete damage states during its life cycle. These damage stages can be classified by the states "new", "nominal", "minimum", and "complete exhaustion of wear-out" introduced in Figure 2.4. To capture the damage progress, these methods are mainly based on Markov models, which include sequential aspects of a given time series. Therefore, the Markov assumption expresses the transition from the paradigm "two consecutive events or states are independent from each other" to an estimation of the underlying connection of both events [Bis13, p. 607]. Based on this assumption, sequential processes as in the field of speech recognition or damage progression are modeled. The Markov models and especially its extension *Hidden semi-Markov Model* (HsMM) are introduced in more detail in Section 3.3.2.

[MTMZ12] and [LBC15] apply this approach to forecast the RUL of bearings. The core of this method is depicted in Figure 2.12. This example represents three different degradation courses of three bearing life cycles. Each bearing i remains for a specific time in states $S^i_{1:N}$. One assumption of Markov models is that every examined component's life cycle starts and ends in the same state A (new state) and Ω (failure, minimum state), respectively. The advantage of this approach is that no specification of a failure threshold or hazard zone as depicted in Figure 2.9 is necessary, since all runs inevitably end in Ω. If the tested component is expected to be in state S^1_1, its RUL is estimated by summing the sojourn time of every state $S^1_{1:N}$ until failure Ω. By assuming a time distribution in every state, the RUL prediction is also distributed.

The approaches in [MTMZ12] and [LBC15] are introduced in detail in Section 3.7. However, the structure of the Markov models in Figure 2.12 already shows that every bearing dataset that is used for the training process of the model is treated separately. Thus, there is no interconnection between the different damage processes of each bearing.

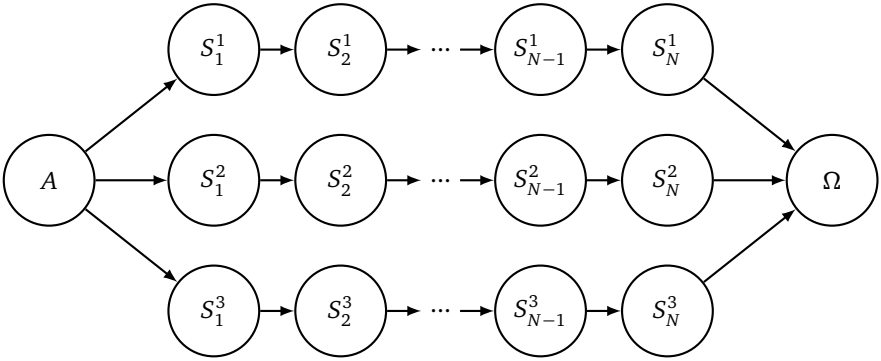


Figure 2.12.: Structure of Markov model applied for bearing damage propagation. Based on [LBC15]

2.4 Evaluation of prognosis results

For a comparison of different PHM approaches, several metrics to assess the prognosis performance are presented in this section. In [SCB+08], [SSSG09], and [SCS+10], several indicators to describe certain characteristics of a prognosis algorithm such as *robustness*, *accuracy*, *precision*, *convergence* or *trajectory*, are summarized and introduced. Thus, this section is based on these publications.

One basic metric is the prediction error

$$\Delta_l(i) = r_l^*(i) - r_l(i), \tag{2.1}$$

where $r_l^*(i)$ is the correct RUL of unit l in prediction step i and $r_l(i)$ is the predicted RUL. Several metrics are derived from the prediction error. The mean absolute percentage error (MAPE) is defined as

$$\text{MAPE}_l = \frac{1}{N_p} \sum_{i=1}^{N_p} \left| \frac{100\Delta_l(i)}{r_l^*(i)} \right| \tag{2.2}$$

for N_p numbers of predictions of unit l. One advantage of this metric in comparison to the prediction error is that errors of late predictions near failure are more penalized. Other accuracy metrics are based on false positives (FP) and false negatives (FN). Similarly to the terms of Table 2.2 in the context of diagnosis, the terms true or false describe the correctness of the predicted RUL; the options positive and negative characterize the predicted confirm or refuse of the hypothesis "*The component will be failed at the predicted time of the RUL*". Thus, a FP indicates a *false alarm* and FN represents a *missed alarm*. The metrics are expressed as

$$\begin{aligned} \text{FP}_l(i) &= \begin{cases} 1 & \text{if } \Delta_l(i) > t_{FP} \\ 0 & \text{otherwise} \end{cases} \\ \text{FN}_l(i) &= \begin{cases} 1 & \text{if } -\Delta_l(i) > t_{FN} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{2.3}$$

where t_{FP} and t_{FN} are predefined constants in order to accept a certain deviation of the actual RUL.

The sample standard deviation (SSD) is applied to evaluate the precision of the predictions. Thus, it is defined as

$$\text{SSD}_l = \sqrt{\frac{\sum_{i=1}^{N_p} \left(\Delta_l(i) - \mu_{\epsilon,l}\right)^2}{N_p - 1}}, \tag{2.4}$$

where $\mu_{\epsilon,l}$ is the mean of the prediction error of the lth unit.

For the description of the prognostic convergence, two metrics are common. The prognostic horizon (PH) expresses the time span, where a predetermined accuracy

constraint α is permanently fulfilled until End of Life (EoL). Thus, the condition for PH is formulated as

$$\begin{aligned} \mathrm{PH}_l &= \frac{N_p - i}{N_p} \\ \text{s.t.} \quad i &= \min\left\{ j \middle| \left(j \in [i, N_p] \right) \wedge \left(r_l^*(j) - \alpha t_{PS} \leq r_l(j) \leq r_l^*(j) + \alpha t_{PS} \right) \right\} \end{aligned} \tag{2.5}$$

where t_{PS} is the total prediction time span calculated with respect to the component's entire life cycle t_{EoL} and the fault detection time t_p, so that $t_{PS} = t_{EoL} - t_p$. One further requirement for PH is suggested in [SCS+10]: as mentioned in Section 2.3.4, besides the mean of the predictions, the prediction uncertainties in particular are of great interest. Thus, the PH is combined with the predicted RUL distribution by analyzing the probability mass within the accuracy bound; if the probability mass is less than a given margin ξ, the conditions for PH are not met. Hence, the constraints of Equation 2.5 are extended by

$$\text{s.t. } i = \min \left\{ j \middle| \left(j \in [i, N_p] \right) \wedge \int_{b^-}^{b^+} p(r_l(i)) dr \geq \xi \right\} \tag{2.6}$$

with the predicted RUL probability density $p(r_l(i))$ and the lower accuracy bound limit $b^- = r_l^* - \alpha r^*(i = 1)$ and analogous the upper accuracy bound $b^+ = r_l^* + \alpha r^*(i = 1)$ in prediction $i = 1$. The margin ξ is the minimum acceptable probability mass within the accuracy bounds; this is set to $\xi = 0.5$ throughout this thesis.

A new metric which focuses only on the prediction uncertainty can be derived from this constraint. By calculating the mean probability mass within the accuracy bound (MPAB) for one unit l, both the prediction accuracy and the spread of each RUL prediction is assessed. The MPAB is formulated as:

$$\mathrm{MPAB}_l = \frac{1}{N_p} \sum_{i=1}^{N_p} \int_{b^-}^{b^+} p(r_l(i)) dr \tag{2.7}$$

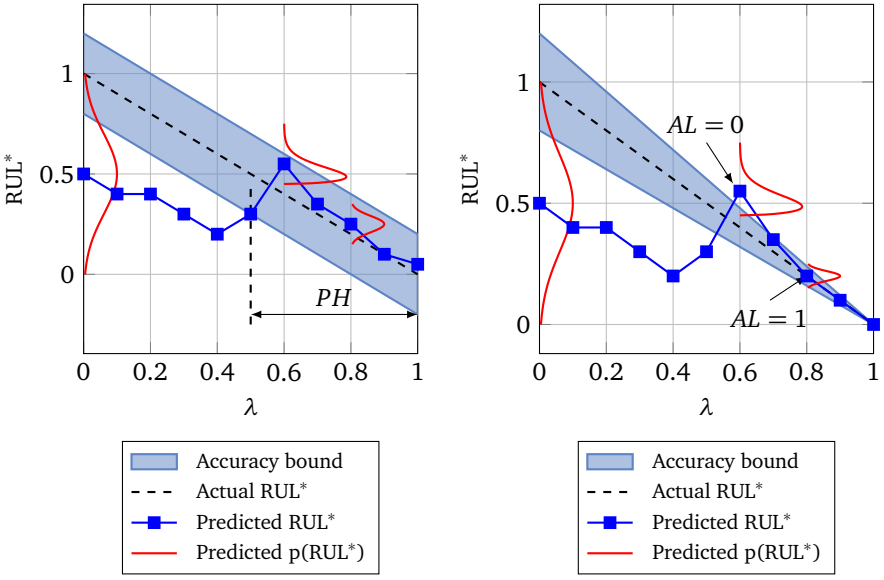
Another common metric which can be used to assess primarily the prognostic accuracy, but also to value the convergence, is the α-λ accuracy (AL), which is defined as:

$$\mathrm{AL}_l(i) = \begin{cases} 1 & \text{if} \quad r_l^*(i)(1-\alpha) \leq r_l(i) \leq r_l^*(i)(1+\alpha) \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

This metric can also be linked to the predicted RUL probability. As suggested in [SCS+10], a second condition analyzes whether the probability mass within the accuracy bound exceeds 50 %. Otherwise, the value for $AL_l(i)$ is null.

A graphical interpretation of PH, MPAB, and AL can be derived from Figure 2.13. Beside the normed actual RUL* in dashed lines, the results of normed example prognoses are plotted in blue. Both are mapped on λ, which is the normed life span of the component after fault detection from $\lambda = 0$ (t_p) to $\lambda = 1$ (t_{EoL}). In addition, the RUL PDF is plotted. In the case of MPAB, the intersections between these distributions and the accuracy bound of Figure 2.13a are evaluated.

The other two metrics can be distinguished by the accuracy bounds. Whereas PH is determined with reference to a constant accuracy constraint, the α-λ accuracy is calculated by a cone-shaped bound. Another distinction is that PH is obtained by finding the earliest point that the prognoses remain within the selected error bound; in contrast, *AL* determines whether the constraint of Equation 2.8 is fulfilled in time step i. Therefore, *AL* is used for accuracy assessment, but the evolution of *AL* over the life cycle also rates the convergence.



(a) Prognostic horizon (PH) and MPAB

(b) α-λ accuracy (AL)

Figure 2.13.: Metrics for the determination of prognostic accuracy and convergence

The final characteristic of a prognosis algorithm is robustness. One possibility is to analyze the robustness to a varying amount of training data and to examine the effects on the abovementioned metrics. If the impact of a varying number of training sets is small, robustness to inherent training data outliers is given.

2.5 Economic assessment of PHM systems

One goal of this thesis is to connect the results of the proposed PHM algorithm with the current maintenance processes with regard to cost benefits. A tool to assess the implementation of predictive maintenance (PM) in a maintenance, repair, and overall (MRO) company in civil aviation is presented in [Käh17]. Therefore, the basics of the implementation are introduced in detail in this section.

The motivation for this tool is that MRO companies have already identified the need for intelligent maintenance, but the introduction of PM for example lacks evaluation methods to assess its benefits in aircraft maintenance. The risks due to implementation costs or consequential costs because of prediction errors are also difficult to estimate. [Käh17, p. 38]

Thus, two objectives are derived: the implementation of a simulation tool to capture today's maintenance process, and a cost-benefit analysis of PM in MRO companies in civil aviation. Thereby, the focus is on components that are maintained correctively, since the expected potential for improvement due to PM is higher than for predetermined maintained components [Käh17, p. 43].

Within the simulation, three different models are applied. These models are interconnected and capture the influences between aircraft operations and aircraft maintenance. The *aircraft operation model* represents the real world flight schedule. Thus, the current state of the examined aircraft (on ground, available for maintenance, etc.) is provided, based on deterministic flight schedule data. The *aircraft maintenance model* includes the sequences of maintenance actions triggered by events. This model is based on maintenance standards, deterministic maintenance history data, and the expertise of maintenance personnel. The final model covers *event initiation*, i.e. induction of maintenance actions. Whereas current initiation is simulated on the basis of historical data (maintenance logbooks, maintenance standards etc.), the prediction-based event trigger in the case of PM is simulated by certain input parameters. These parameters describe the prognostic performance, and are therefore metrics from Section 2.4, such as the PH. The other parameters are the investment costs for PM implementation C_P, and two metrics to assess the prediction accuracy by means of the FP and FN. Instead of the prediction FP and FN of Equation 2.3, where the hypothesis "*the component will be failed at the predicted time of the RUL*" is tested in every prediction, another approach, which

also includes the time for maintenance actions Δt_{ma} is selected in [Käh17, p. 34]; as soon as a prediction goes below a specific Δt_{ma}, a maintenance event is initiated. Thus, if for example the median of the estimated RUL probability distribution falls below Δt_{ma}, an FP is generated. In contrast, when Δt_{ma} is not crossed at all, the prediction set is assessed as FN. A TP (true positive, i.e. the correctly predicted failure) is generated only if the true RUL is estimated accurately (especially at the end of life). The metrics that are based on these values are false negative rate (FNR) and specific false discovery rate (SFDR) as defined by [Käh17, pp. 56–85]:

$$\text{FNR} = \frac{\text{FN}}{\text{FN} + \text{TP}} \tag{2.9}$$

$$\text{SFDR} = \frac{\text{FP}}{\text{FP} + \text{FN} + \text{TP}} \tag{2.10}$$

By varying these input parameters within Monte-Carlo simulations, the overall annual costs, consisting of operating costs (labor, aircraft purchase, fees), maintenance costs (material, logistics), and delay and cancellation costs (dependent on delay time) are estimated. Hence, current costs with corrective maintenance and the total costs with PM can be compared as depicted in Figure 2.14. This plot is a result for a constant PH= 300 min and $C_P = 17{,}500\,€/\text{a}$ for variable FNR and SFDR, and it shows the break-even of current and prediction-based maintenance. As expected, in the case of low prediction errors (i.e. small FNR and SFDR) the total costs for predictive maintenance are smaller than for corrective maintenance.

Thus, by connecting the results of the PHM algorithm in this thesis with the outcome of the cost-benefit tool of [Käh17], in Section 5.5.2 it is inferred whether or not the application of the introduced algorithm is beneficial. However, it must be considered that the use cases of [Käh17] and the examined component within this thesis differ. Hence, actual effects on the maintenance process cannot be derived, but trends concerning cost-benefits are determined.

2.6 Delimitation to other approaches

The introduction to the field of PHM identified state-of-the-art approaches, but also current problems. Thus, this section summarizes these challenges and how they were faced in this thesis using several new approaches.

One severe problem in data-driven PHM for real applications is that mostly only unlabeled data is available; the actual degradation of the examined component remains unknown. Especially in the field of mechanical degradation, the dispersion of e.g. cracks or pits cannot be measured directly in the process. Thus, features to express the current state of the component are difficult to identify, since the

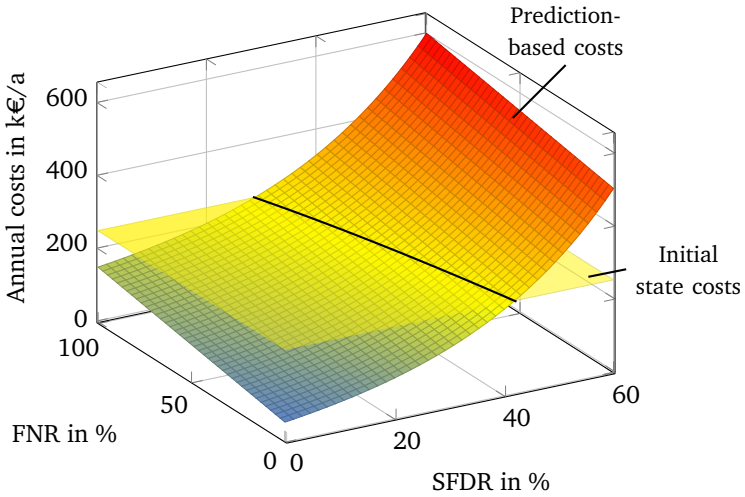


Figure 2.14.: Total cost example for PH $= 300\,\text{min}$, $C_P = 17{,}500\,€/\text{a}$. Based on [Käh17, p. 132]

characteristic features also vary within different degradation processes. A new approach in the field of feature generation and reduction that is suggested in this thesis is able to select the relevant features automatically.

Another challenge results from the behavior of features during the degradation process. The idealized monotonic increase is usually not measured during run to failure trials, due to recovery processes or because the selected signal type does not represent the actual health state. One method to overcome this problem is the use of Markov models based on multiple observations. However, established methods in this field lack the ability to classify similar degradation processes. Within these approaches, each training run is treated independently. Consequentially, these algorithms encounter problems if a tested degradation process differs from the applied training data. A combination of all training data in one damage propagation model, as suggested in Figure 2.15, might benefit from more information about the multitude of possible damage courses and the underlying stochastic of the degradation process. In comparison to Figure 2.12, several damage states of different runs are not separated, but can be reached by the damage states of other runs.

Besides the health index as an input for the damage propagation model, the endured load is also analyzed as an additional input. Due to another signal that af-

fects the damage, the determination of current and future degradation is expected to be more accurate.

One further novelty is the evaluation of the proposed PHM algorithm with the motor current of an induction machine. This decision is motivated in cases of difficulties to plug vibration sensors for example in certain fields of applications with a harsh environment. Another use case of the analysis of motor current instead of vibration are applications with application-dependent vibration, which superimposes the fault signals. Therefore, a new test rig design was realized for the generation of degradation data.

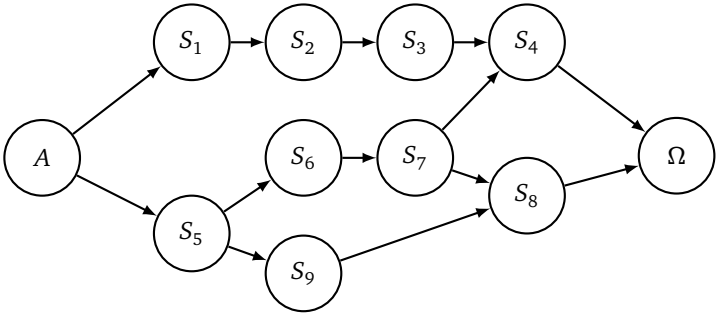


Figure 2.15.: Example of assumed damage propagation sequence

3 Basics of the proposed PHM algorithm

The concept of a new PHM algorithm, which is the focus of this thesis, is presented in this chapter. Firstly, an overview of the entire algorithm is presented, separated into training and test phases. This overview provides the structure for the subsequent sections, which also complies with the framework presented in Section 2.3. Firstly, the signal processing methods are explained in Section 3.2. Based on the selected feature subset $\mathbf{\Phi}$, which is the output of this section, models are trained. Here, the focus lies on the introduction of Hidden semi-Markov Models (HsMM), which are at the center of the fault diagnosis and prognosis steps. This includes the fundamentals of HsMMs (Section 3.3.2), the optimal selection of model parameters (Section 3.3.3), and the implementation of a model with the structure shown in Figure 2.15. The trained models then form the basis for fault diagnosis during the test phase. This step consists of the estimation of the current damage state (fault identification) and the selection of the most suitable model. Based on these two outputs, the fault prognosis is initiated. The two concepts of Remaining Useful Life (RUL) estimation are established in Section 3.5.1. Since the concepts for RUL estimation result in non-Gaussian probability distributions, a sample approach in the field of Markov Chain Monte Carlo (MCMC) simulations is applied.

After the concept survey, the proposed PHM algorithm is verified in Section 3.6. Here, the different modules are tested with state-of-the-art verification methods in the field of PHM as well as computer science. After the concept verification, a comparative algorithm is introduced in Section 3.7. Here, the basic idea presented in [MTMZ12] forms the foundation of this algorithm.

Throughout this chapter, datasets of degrading bearings that are investigated as use case in Chapter 4 and Chapter 5, are applied as examples in order to demonstrate the performance of each module. However, similar signals are assumed to be collected in the condition monitoring of gears, pumps, or other rotary machines so that the proposed PHM algorithm is also suitable for these application fields.

3.1 Overview of the entire PHM algorithm

The PHM algorithm is divided into off-line and on-line phases. Whereas the off-line step consists of the training process of the applied models by means of training datasets, the predictions for new test sets are provided during the on-line step. These phases are illustrated as flow charts in Figure 3.1 and Figure 3.2. The selected colors and terms follow the flow chart of Figure 2.5 to identify particular steps such as data generation, diagnostics, or prognostics.

The training process in Figure 3.1 begins with a data conversion of the recorded datasets. This step is application-dependent according to the examined condition monitoring (CM) signal (e.g. the motor current I_{ai}) and the data acquisition platform (e.g. ControlDesk by DSpace). The aim of this step is to transform the CM signal into a more feasible data format. This includes steps such as conversion from a recorded current sensor signal given in volts into the corresponding signal I_{ai} in amperes. In the case of rotary components, these examined CM signals are sinusoidal waves which are sampled for a given measurement window. The entirety of all I_{ai} throughout one component's life cycle from $i = 1,2,...,T$ builds the block *Data run* depicted in Figure 3.1. Each current signal I_{ai} of time step i is then filtered by a notch filter to reduce the amplitudes near the supply frequency and its harmonics. The signals are then transformed into the frequency domain represented by the power spectral density (PSD) of the signal. In order to achieve a holistic model which covers the degradation progress presented in Figure 2.15, where similar damage states can be reached during different degradation processes, these PSD spectra must be comparable for all datasets. However, since the amplitudes of different datasets vary considerably, a normalizing step with respect to a predefined nominal state is necessary. In this way, the features for all training datasets from run 1 to N_{td} are generated and can be joined into one normed feature matrix $\hat{\boldsymbol{S}}_{\boldsymbol{YY}}$, which comprises the damage progress of N_{td} different datasets.

After the feature generation step, the matrix $\hat{\boldsymbol{S}}_{\boldsymbol{YY}}$ contains several features with no or similar information. Hence, to reduce the number of features and to simultaneously increase the information content, the two criteria *mutual information* and *standard deviation* of single features are evaluated during the feature selection step. A discretization according to the amount of continuous features $\boldsymbol{\Phi}_c$ into a predefined number of levels further decreases influences due to noise. These discretization levels are saved for the test procedure. The discrete features form the multi-dimensional feature matrix $\boldsymbol{\Phi}$ for fault diagnosis.

For a first classification of the underlying damage states and especially for dimension reduction, the feature matrix $\boldsymbol{\Phi}$ is clustered by the K-means algorithm. By means of the clustering, similar damage conditions in different runs are identified.

In addition, the clustering transforms the unlabeled data $\boldsymbol{\Phi}$ into the labeled clusters $\boldsymbol{C}$; both, $\boldsymbol{\Phi}$ as input vector and $\boldsymbol{C}$ as target vector are applied in order to train the classification algorithm K nearest neighbor (KNN) to classify new data. Finally, the cluster vector $\boldsymbol{C}$, which captures the short term damage classes $\boldsymbol{C}$, serves to train several Hidden semi-Markov Models (HsMM) to determine the long term damage states $\boldsymbol{S}_i$ of Figure 2.15. Since these models are used for fault diagnosis (current state of the unit) and for fault prognosis (future state of the unit), the fill color of the process block *Training HsMMs* is both light gray and light blue in accordance with Figure 2.5.

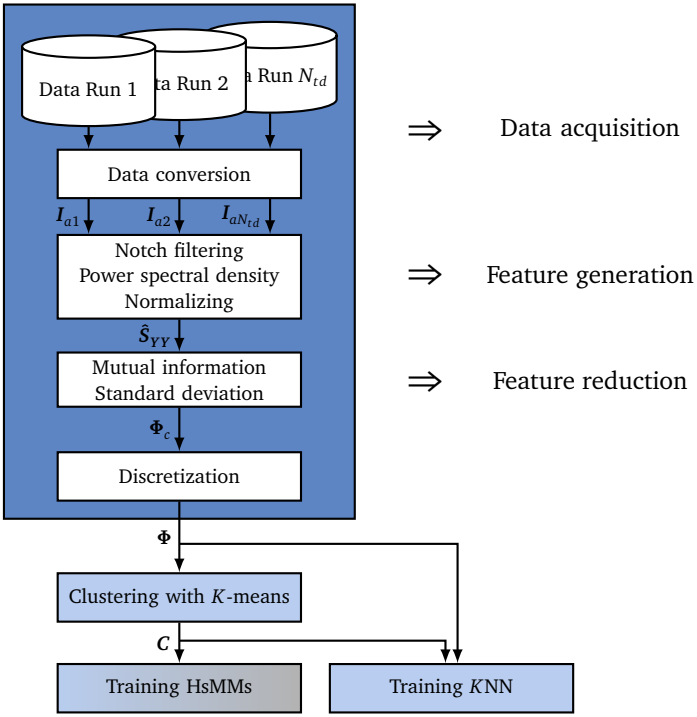


Figure 3.1.: Training process of the proposed PHM algorithm

Based on the outputs of the training process, a new test run is assessed, and is depicted in Figure 3.2. First, the new dataset is passed through the same steps as the signal processing that is part of the training process. Here, several parameters which result from the training process e.g. (the normalizing factors during the feature generation or the selected feature subset of the feature reduction) are

applied. Likewise, the limits of the discretization step identify the current level of each feature. In this way, the feature matrix $\mathbf{\Phi}$ for a new test data run is generated.

The fault diagnosis of the new test dataset begins with the *K*NN model of the training process. This is applied in order to classify the feature matrix $\mathbf{\Phi}$ of the new dataset into the short term damage classes $\boldsymbol{C}$. In addition, the *K*NN provides a score $\boldsymbol{S}_C$ that expresses the likelihood of the features belonging to a certain class, therefore indicating the class probability. Considering both the current short term damage state $\boldsymbol{C}$ and the class probability $\boldsymbol{S}_C$, the current long term damage state $\boldsymbol{S}_i$ is determined by means of the trained HsMMs. As illustrated in Figure 3.2, the output of the training process are several HsMMs, which include the damage sequence of one particular run 1 to N_{td} and also the combination of all sequences in the *HsMM Net*. This net follows the example fault propagation sequence of Figure 2.15. The assumption behind the application of both types of HsMMs is that the algorithm has more options to determine a suitable model m_j that represents the current degradation process for the examined test set. Thus, the outcome of the fault diagnosis is the estimated current damage state $\boldsymbol{S}_i$, the current duration $\boldsymbol{d}_i$ within this state, and the model probability $P(m_j)$, which assesses the fitness of each model.

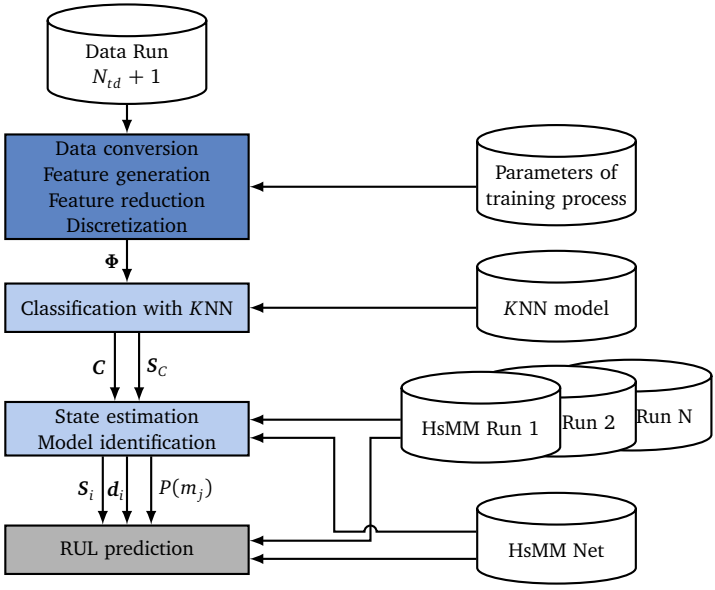


Figure 3.2.: Test procedure of the proposed PHM algorithm for a new test data set

After localizing the current position within the HsMM chains (fault diagnosis), the models are used to proceed until the failure damage state Ω is reached (fault prognosis). The output of this last step is the probability distribution of the remaining useful life (RUL).

Each step that has been briefly introduced in this section, is presented in detail in the subsequent sections. The structure of the sections is analogous to Section 2.3. All necessary parameters are summarized in Section A.5.

3.2 Signal processing

This section presents the determination of feature matrix $\mathbf{\Phi}$ for the subsequent fault diagnosis. Since the data acquisition is application-dependent, it is omitted in this section and introduced in detail in Section 4.2.2.2 and Section 5.1.1. Thus, feature generation is presented first.

3.2.1 Feature generation

At the beginning of the feature generation step, it is assumed that the CM signal is available in the time domain. Since the focus of the application fields is narrowed down to rotary machines, several frequency bands (and their harmonics) that outshine all other amplitudes are filtered out by a Butterworth notch filter. These are for example the supply frequency in the case of the motor current or the rotary frequency in vibration signals, neither of which indicate the damage progress.

The remaining signal is transformed into the frequency domain represented by the PSD spectrum. Other approaches such as the wavelet transformation are ignored, since the examined process is expected to be stationary, and therefore the consideration of time-frequency domain methods as introduced in Figure 2.6 are not necessary.

By plotting all PSD spectra from new state to failure in one plot, the damage progress over the entire life cycle of a component is illustrated in Figure 3.3. The frequency range around 30 Hz in particular increases strongly as the damage proceeds. However, this growth is not monotonic, since all amplitudes fall at approximately 40 h. The influences of the abovementioned notch filter are also visible, since the amplitudes at 50 Hz are significantly reduced.

The amplitudes of PSD spectra $\hat{\mathbf{S}}_{YY}$ are applied as features. However, since the amplitude values of different runs vary greatly, these features are normalized with respect to the amplitudes at the beginning of each trial. The time span between 1 h and 3 h of each run is expected to be the nominal state, so that all features are

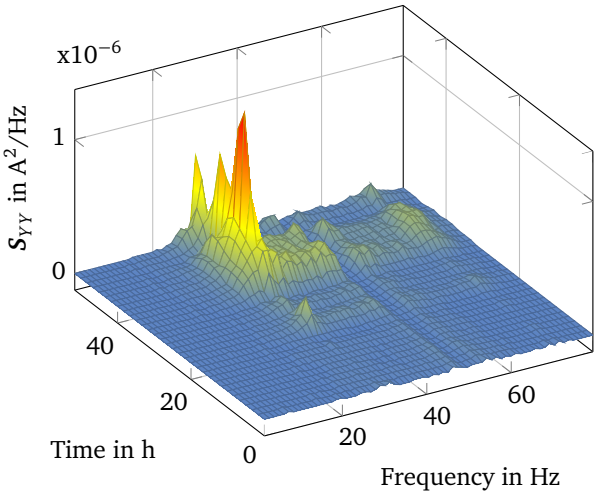


Figure 3.3.: Power spectral density spectra over a component's life cycle

scaled according to the 95th percentile of the amplitudes within this time period. Besides this normalizing step outliers which exceed the 95th percentile of the amplitude of each feature, are also set to the 95th percentile. The resulting normed amplitudes $\hat{\boldsymbol{S}}_{YY}$ are then transferred to the feature reduction step.

3.2.2 Feature reduction

As depicted in Figure 3.3, many features display similar behavior. Thus, the number of features can be reduced without loss of information. One criterion to assess the amount of information that two different signals possess is the *mutual information criterion*. This is defined in accordance with [Bis13, pp. 55–58] as:

$$I[\boldsymbol{x},\boldsymbol{y}] = -\iint P(\boldsymbol{x},\boldsymbol{y}) \ln\left(\frac{P(\boldsymbol{x})P(\boldsymbol{y})}{P(\boldsymbol{x},\boldsymbol{y})}\right) d\boldsymbol{x}\, d\boldsymbol{y} \tag{3.1}$$

The value $I[\boldsymbol{x},\boldsymbol{y}] = 0$ indicates independence between the two signals. The goal is to discard one of two features with high mutual information. Since this criterion only provides a metric for the comparison of two signals, an additional criterion is necessary to determine which feature is folded. Therefore, the standard deviation of each signal is analyzed, so that the feature with the higher standard deviation is preferred.

For the final feature reduction, the standard deviation of all remaining features is analyzed and only the n features with the highest standard deviation are selected. The results for two runs (run 1 from 0 to 50 h and run 2 from 50 to 216 h, both separated by a dashed line) are illustrated in Figure 3.4. For a better visualization, both runs are plotted consecutively. They show that for different runs, diverse features are necessary in order to capture the degradation process of each component. In addition, the restriction of outliers (above 95th percentile), as mentioned in the previous subsection is visible, since both features run into saturation and are cut off at a certain feature amplitude. All n *continuous* features are summarized in one matrix $\mathbf{\Phi}_c$.

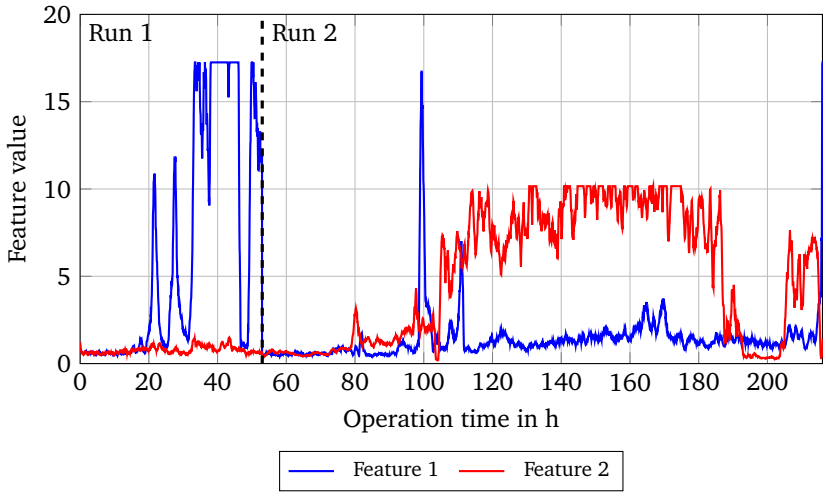


Figure 3.4.: Two features of different runs after feature reduction by means of mutual information criterion and standard deviation

3.2.3 Discretization

Discretization according to the amount of features is necessary, since certain features have a low signal-to-noise ratio, and in addition vary in their amplitudes. Since the goal of the next step is to summarize these data points into classes, the subsequent clustering algorithm has problems as it is based on the distance between the data points [Alp10, p. 163]. Thus, discretization supports this process, as it reduces the altering length among features. The signals of $\mathbf{\Phi}_c$ are firstly low-

pass filtered and then summarized into m discrete levels in accordance to their amplitude. The results for the features of Figure 3.4 are plotted in Figure 3.5 in the case of $m = 5$ discrete levels. All n features are summarized in the final feature matrix $\mathbf{\Phi}$, which is transferred to the fault diagnosis step.

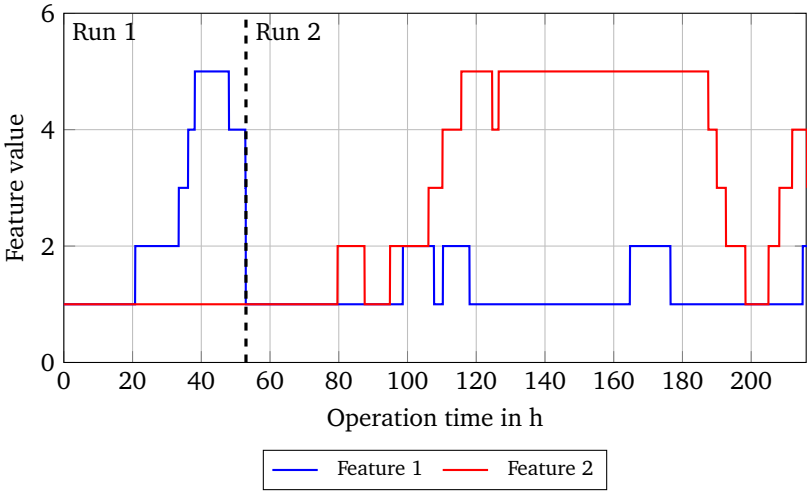


Figure 3.5.: Discrete features of two different runs

3.3 Model training

Two different kinds of models are trained during the training process (see Figure 3.1): *K Nearest Neighbor* (*K*NN) based on the results of clustering by *K-means* for *data classification* and *Hidden semi-Markov Model* (HsMM). Both methods are introduced in this section. As depicted in Figure 3.1, the training of HsMMs cannot be assigned to either fault diagnosis or fault prognosis alone, since it is used in both steps. Thus, this section varies from the structure of Section 2.3.

3.3.1 Data classification

One challenge in data-driven PHM approaches is the handling of unlabeled data: since the actual degradation state remains unknown, assignment of the current damage level to the actual severity of the fault is difficult. Hence, one goal of the

proposed PHM approach is to train an algorithm that self-assigns the degradation states into damage classes based on the feature matrix $\mathbf{\Phi}$.

Cluster algorithms are able to transform unlabeled data into labeled data. As introduced in Section 2.3.3.2, one popular method is the K-means algorithm. Reasons for its widespread application are that it does not require any prior knowledge about the provided signal, and its simplicity. The aim of the algorithm is to partition the data points of a signal into K clusters. The clusters are characterized by a point cloud, where the inter-point distance is small in comparison to the distance to points outside the cluster. The clusters are defined by its center $\boldsymbol{\mu_k}$ so that the goal is to find K cluster centers $\boldsymbol{\mu}_k$ such that for a given data input vector $\boldsymbol{x}$ the objective function

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2 \tag{3.2}$$

is minimal. Here, N is the number of input data points and r_{nk} is a binary indicator to describe whether the data point x_n is assigned to cluster k ($r_{nk} = 1$) or not ($r_{nk} = 0$). Thus, by iteratively optimizing r_{nk} and $\boldsymbol{\mu}_k$ the objective function J is minimized. One crucial step during the parametrization of the algorithm is the selection of a distance measurement between the input vector and cluster centers. In Equation 3.2, the Euclidean distance is chosen. The entire algorithm with different suggested distance measurements is introduced in [Bis13]. [Bis13, pp. 424–428]

Besides the choice of a suitable distance measurement, another challenge of this algorithm is to find a suitable value for K. In [Agg15, pp. 196–198], several criteria for the assessment of the internal validity are introduced. Among others, the *Silhouette coefficient* is presented, which determines both the distance of points within a cluster and the distance between two clusters. Therefore, a large distance between the clusters on one hand and a small point cloud within a cluster on the other lead to a high Silhouette coefficient, which indicates highly separated clustering. Thus, it is applied as an optimization criterion to determine K, which is discussed in more detail in Section 3.3.3.

A result of clustering the feature matrix $\mathbf{\Phi}$ into clusters $\boldsymbol{C}$ with K-means is plotted in Figure 3.6. Every feature of run 1 is given analogous to Figure 3.5 from damage levels 1 to 5 in the upper part of the figure. In this example the different combinations of features and their damage levels creates the necessity for 16 clusters in order to separate each damage class.

One drawback of K-means is that the results are not deterministic, but depend on the initial position of $\boldsymbol{\mu}_k$. One possible way to face this challenge is to begin the clustering with different initial positions in order to avoid local minima. The

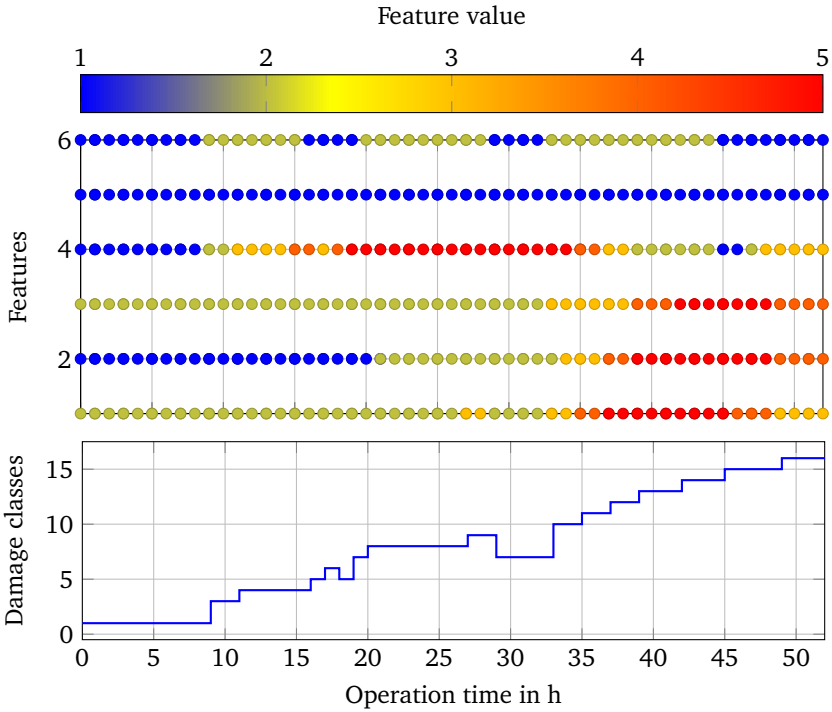


Figure 3.6.: Clustering results in combination with the applied features of one run

configuration with the lowest value for the objective function J and the highest Silhouette coefficient are used as final clusters.

The important outcome for the subsequent model training is the reduction of dimensionality from feature matrix $\boldsymbol{\Phi} \in \mathbb{R}^{k \times n}$ to damage classes $\boldsymbol{C} \in \mathbb{R}^{k \times 1}$ for n numbers of feature and k time steps. Both signals are used to train a classification algorithm. A wide range of methods (support vector machines with different kernels, decision trees, several KNN approaches) was evaluated. The selected metric to identify the most suitable classification algorithm was thereby the misclassification rate, which is represented by the confusion matrix introduced in Table 2.2. The outcome of this evaluation is presented in Section A.1. It shows that all classifiers were able to assign new test data points to the correct damage classes with no misclassifications. Thus, the KNN algorithm is applied due to its simplicity and temporal benefits compared to the other approaches.

As the name *K nearest neighbor* implies, the classification of new data points $\boldsymbol{x}$ with the *K*NN approach is made according to the K surrounding training points of the feature matrix $\boldsymbol{\Phi}$. With reference to the corresponding clusters $\boldsymbol{C}_k$, the posterior probability of test point $\boldsymbol{x}$ lying within cluster $\boldsymbol{C}_k$ is:

$$P(\boldsymbol{C}_k|\boldsymbol{x}) = \frac{K_k}{K} \tag{3.3}$$

Here, K is the preselected number of nearest training points and K_k is the number of training points that belong to cluster $\boldsymbol{C}_k$. Thus, for every new data point $\boldsymbol{x}$ a class probability $\boldsymbol{S}_C = P(\boldsymbol{C}_k|\boldsymbol{x})$ is given. Again, a distance measurement similar to the K-means algorithm must be selected to identify the nearest training data points. Analogous to K-means, the Euclidean distance is chosen. The number of nearest neighbors K is again selected with respect to the lowest misclassification rate during a holdout-validation in the training phase. The results summarized in Table A.1 show that $K = 5$ is a suitable choice. A more detailed introduction into this algorithm is provided in [Bis13]. [Bis13, pp. 124–127]

By means of the *K*NN, which is trained with the results of K-means, new test runs are classified, so that the number of features is reduced from n to 1, and through this similar damage classes are identified. This one-dimensional signal then offers the opportunity to further cluster these damage classes into damage states by training Hidden semi-Markov Models (HsMMs), which are introduced in the next section.

3.3.2 Fundamentals of Hidden semi-Markov Models

The Hidden Markov Model (HMM) approach was introduced in [BP66]. One of the first applications of this was in the field of speech recognition, introduced in [Rab89]. Later, further application fields such as handwriting recognition [NSS96], motion capturing [YOI92], or encoding of human DNA [HE96] were added. All applications have the existence of *sequential data* in common.

The core of an HMM is a Markov model. The assumption of Markov is that at any time a system is in one of N distinct states S_1,...,S_N. From the current state $q_t = S_i$ in time step i, the system passes on to either another state S_j or it remains in the initial state S_i, which is captured by the *transition probability*. Here, the order of the Markov model determines, how many predecessor states are considered for the transition probability. For a first order Markov chain, the general formulation of the transition probability is cut off after the last state q_{t-1}, so that it is defined as

$$a_{ij} = P(q_t = S_j|q_{t-1} = S_i, q_{t-2} = S_k,...) = P(q_t = S_j|q_{t-1} = S_i) \tag{3.4}$$

in cases where the transitions are time independent. By additionally determining the initial state with the probability $\pi_i = P(q_1 = S_i)$, an entire process can be modeled. [Rab89, pp. 258–259]

The assumption of Markov models is that the states S_i are directly observable. This restriction is softened by including cases where the observations are probabilistic functions of the states. Here, the actual state transitions are not observable (hidden), so that this extension of Markov models is the abovementioned HMM. Thus, in addition to a state transition matrix $\boldsymbol{A} = a_{ij}$ and an initial state probability vector $\boldsymbol{\pi} = \pi_i$, the HMM is characterized by a further emission matrix $\boldsymbol{B}$ with its entries

$$b_{jk} = P(v_k | q_t = S_j) \tag{3.5}$$

where v_k is the kth observation ($k = 1,...,M$) of the observation sequence O_t in time step t. By means of $\boldsymbol{B}$, the observations are now connected to the hidden states S_j. The sequential observations are assumed to be statistically independent. Thus, an entire HMM is defined by its parameters $\boldsymbol{\lambda} = (\boldsymbol{A},\boldsymbol{B},\boldsymbol{\pi},N)$, where N is the number of states. [Rab89, pp. 259–261]

Three different problems arise from this form during the training of HMMs. They are formulated in [Rab89, p. 261] as:

Problem 1 How is the probability of an observation sequence $P(\boldsymbol{O} = O_1 O_2 ... O_T | \boldsymbol{\lambda})$ efficiently computed for a given model $\boldsymbol{\lambda} = (\boldsymbol{A},\boldsymbol{B},\boldsymbol{\pi},N)$?

Problem 2 How is a corresponding state sequence $\boldsymbol{Q} = q_1 q_2 ... q_T$, which is optimal to model the underlying process, derived from the observation sequence $\boldsymbol{O} = O_1 O_2 ... O_T$?

Problem 3 How are the model parameters $\boldsymbol{\lambda} = (\boldsymbol{A},\boldsymbol{B},\boldsymbol{\pi},N)$ adjusted to maximize $P(\boldsymbol{O}|\boldsymbol{\lambda})$?

Problem 1 is also called the evaluation problem, since it gives a hint about the model fitness to a given observation sequence represented by $P(\boldsymbol{O} = O_1 O_2 ... O_T | \boldsymbol{\lambda})$. One approach to solving this problem is the forward step of a *Forward-Backward* procedure. By inductively propagating a forward variable $\boldsymbol{\alpha}_t(i)$ from time step t to the end T, depending on the model parameters in $\boldsymbol{\lambda}$ the probability of the observation sequence $P(\boldsymbol{O}|\boldsymbol{\lambda})$ is determined. Here, $\boldsymbol{\alpha}_t(i)$ is defined as:

$$\boldsymbol{\alpha}_t(i) = P(O_1 O_2 ... O_T, q_t = S_i | \boldsymbol{\lambda}) \tag{3.6}$$

The entire algorithm containing the mathematical formulations of this forward step is presented in Section A.2. [Rab89, pp. 262–263]

Problem 2 leads to an optimization problem to determine the best fitting state sequence $\boldsymbol{Q} = q_1 q_2 ... q_T$ of the underlying process. For this, a backward variable $\boldsymbol{\beta}_t(i)$ is defined from time step T to the end t analogous to $\alpha_t(i)$, but starting at the end of the process. By combining the information of both variables $\boldsymbol{\alpha}_t(i)$ and $\boldsymbol{\beta}_t(i)$, the most probable state q_t is derived. Besides the Forward-Backward algorithm, another approach to solving this problem is the Viterbi algorithm, which is introduced in [Rab89, p. 264]. [Rab89, pp. 263–264]

The final problem includes the re-estimation of the model's parameters $\boldsymbol{\lambda}$ to maximize $P(\boldsymbol{O}|\boldsymbol{\lambda})$ from problem 1. One state-of-the-art method for this re-estimation is the *Baum-Welch* algorithm. Here, all parameters are iteratively changed according to a *Lagrange optimization*, which includes the changes of $P(\boldsymbol{O}|\boldsymbol{\lambda})$ with respect to the selected parameters. Thus, an optimal set of parameters $\boldsymbol{\lambda}$ is chosen. The basic formulas are also provided in Section A.2.

By means of these three steps, an HMM, which is able to predict the next states on basis of the trained runs, is set up. In the case of HMM, these state transitions, and thus also the duration time in a state S_i, depend only on the entries of the transition matrix $\boldsymbol{A} = a_{ij}$. An example of a state sequence in the case of HMM is presented in Figure 3.7a.

The duration time within a state S_i of an HMM is mathematically equivalent to an exponential duration density given by:

$$P_i(d) = (a_{ii})^{d-1}(1 - a_{ii}) \tag{3.7}$$

where a_{ii} is the probability of remaining in state S_i. In [Rab89, pp. 269] the assumption of an exponential duration density is denoted as inappropriate for the majority of physical signals. Thus, [Fer80] introduces a method based on the structure of HMM, but with an explicit duration density. With this assumption, the underlying model for the state sequence is no longer captured by a Markov model, but by a *semi-Markov* model. The extension to hidden states is consequentially called *Hidden semi-Markov models* (HsMM). The difference between HMM and HsMM is illustrated in Figure 3.7: the structure of both HMM and HsMM is the same, but the self transition is changed from a constant self-transition captured by a_{ii}, which results in the duration density given by Equation 3.7, to an adjustable duration density $P_i(d)$. Instead of the exponential distribution of Equation 3.7, other probability distributions such as Gaussian or Gamma distributions are selectable.

The extension from HMM to HsMM demands changes for the algorithm, which are presented in Section A.2. The parameters to define an HsMM consists of $\boldsymbol{\lambda} = (\boldsymbol{A},\boldsymbol{B},\boldsymbol{\pi},\boldsymbol{D},N)$, where $\boldsymbol{D}$ is the duration matrix containing the duration densities $P_i(d)$. These models are the core of the proposed PHM algorithm.

The implementations for model generation are based on the toolbox by Yu and Kobayashi, which is introduced in [YK06]. This algorithm forms the basis for the

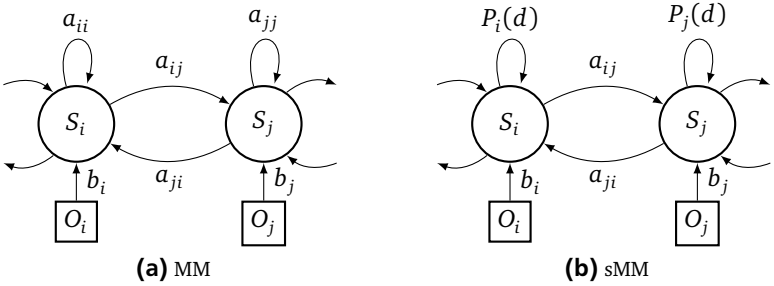


Figure 3.7.: Difference between HMM and HsMM. Based on [Rab89, p. 269]

mathematical formulations in Section A.2. Some extensions are necessary, since the classification results of *K*NN and its uncertainty must be processed during the model setup.

In Figure 3.8, the results of the damage state identification by means of HsMM for the damage classes of Figure 3.6 are presented. Several clusters of the *K*-means algorithm in blue form the final damage states in red. The connection of both signals is captured by the emission matrix $\boldsymbol{B}$ of Equation 3.5.

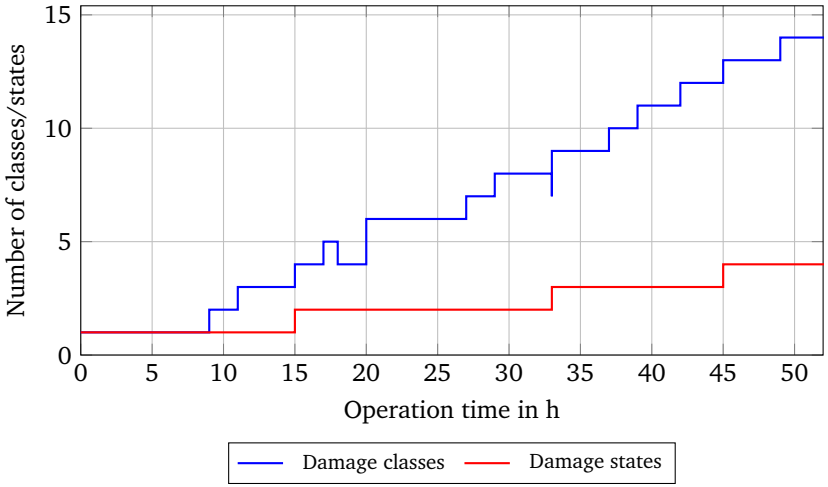


Figure 3.8.: Results of damage state identification with HsMM

3.3.3 Parameter optimization steps

Several parameters are necessary for K-Means and HsMM. Some are determined by the user, for example the amount of training data or the number of features, but the majority of parameters are intended to be chosen autonomously. Thus, suitable optimization criteria are necessary in order to realize a self-adjusting PHM algorithm.

As described in Section 3.3.1, one objective function for the determination of the number of clusters K is the Silhouette value Sil, which is defined for ith data point as [Alp10, pp. 196–197]

$$Sil_i = \frac{D^{out}_{min_i} - \bar{D}^{in}_i}{\max\left(D^{out}_{min_i}, \bar{D}^{in}_i\right)} \tag{3.8}$$

where $\bar{D}^{in}_i$ is the average distance of all data points $\boldsymbol{X}$ inside the corresponding cluster. $D^{out}_{min_i}$ represents the minimum average distance of data points within a cluster to the data points of another cluster. The overall Sil value is the average of every ith data point specific Sil_i. The limits for the Silhouette value are $Sil = 1$ for highly separated clustering and $Sil = -1$ indicates a mixing of data points from different clusters.

In Section A.3, one example of Silhouette values for an increasing number of K is presented. The course of Sil converges for high values of K to one. Since the resulting value of K for $Sil = 1$ is too high, a limit of $Sil = 0.9$ is defined to reduce the model complexity (comparable to the motivation in [Rou87]). Thus, the number of clusters K is determined.

Another important parameter that must be predefined is the number of damage states N for the HsMM. In [Le15, pp. 77–78] the Bayesian Information Criterion (BIC), which is introduced in [Sch78], is suggested to identify N. The BIC value is defined as:

$$BIC = -2\ln(L_\lambda) + N\ln(n) \tag{3.9}$$

Here, $L_\lambda = P(C_1 ... C_T|\boldsymbol{\lambda})$ is the likelihood of $C_1 ... C_T$ generated from an HsMM with model parameters λ and n is the number of data points. One advantage of this criterion is that it includes not only the model accuracy by means of L_λ, but also the model complexity with the second term. Since a low BIC is required, a model with higher complexity is punished.

The number of damage states N is iteratively increased during the training process in order to calculate the BIC value. The final value for N is then selected according to the number of states with the lowest value of BIC.

3.3.4 Training of the net

Based on the results of the previous section, several HsMMs are trained. One novelty in comparison to other approaches in the field of HsMM methods is the application of a model that contains all available information about the damage propagation sequence of all training runs. This model is called *net* throughout this thesis and the theory behind it is illustrated in Figure 2.15. Therefore, all training runs are assessed simultaneously with regard to the damage progress, so that similar damage states are identified. The outcome of this damage state identification also builds the basis for the single HsMMs.

One example of this training process is illustrated in Figure 3.9 for two runs, which are combined in the net. Again, both runs are plotted consecutively for a better visualization. Based on the damage classes from K-means, the damage states are trained during the HsMM setup. Although both components degrade in different ways, since the damage states near end of life vary, the nominal state S_1 is equal in both runs. As presented in Figure 3.2, in addition to the net single HsMMs are also trained with each individual dataset. Thus, the damage state sequences derived by the net training are then separated again for each training run in order to set up these single HsMMs, so that each single HsMM obtains individual parameters $\boldsymbol{\lambda} = (\boldsymbol{A},\boldsymbol{B},\boldsymbol{\pi},\boldsymbol{D},N)$. In the case of the first training run of Figure 3.9, the state transition matrix $\boldsymbol{A}$ would only permit transitions $S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4$ and would then jump into failure state Ω.

The results of Figure 3.9 indicate that both runs show a monotonic rising behavior related to damage states, i.e. no damage states are entered multiple times within one run. This can also be interpreted as a measure for model validity. In contrast, if there are damage states that appear multiple times during one run, cycles within the transition of HsMM are produced, which leads to inaccurate predictions. Thus, the presence of these cycles must be avoided, which is not possible for every dataset. In [MTMZ12], these cycles are constrained by counting the maximum number of damage state entries during the training process; this number of entries may not be exceeded during the prognosis step, so that an infinite number of state entries is not possible. This method is also adapted for the postulated approach.

In comparison to Figure 2.15, the two important states *initial state A* and *failure state* Ω are omitted, since both last only for one time step. However, both states are important for the theory of the applied concept. The initial state A indicates that each component starts in the same state regardless of the subsequent degradation process. This state is solely emitting, since it cannot be reached by other damage states. On the other hand, the failure state Ω brings together all different failure

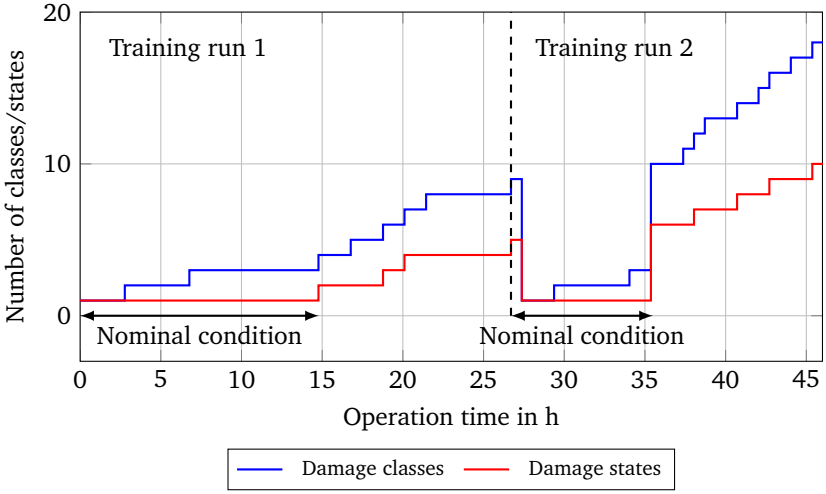


Figure 3.9.: Two different runs used for training of the HsMM net

modes. Thus, this final state is absorbing. In the case of Figure 3.9, the failure state Ω corresponds to S_{11}.

The training process of the net is completed by identifying the nominal states of each training run. In the example of Figure 3.9, the two nominal states are S_1 and S_5, since both states are entered at the beginning of the life cycles. Thus, fault predictions can be firstly initiated after leaving these states.

3.4 Diagnosis step

The aim of fault diagnosis is the detection, isolation, and identification of faults. All three steps are realized by the definition of holistic damage states that all components pass through during their life cycle. Therefore, an incipient fault is detected when the component changes from its nominal state to another. The detection of the fault is the initiation of fault prognosis, as indicated in Section 2.3.4. In addition, different faults are isolated and identified by diverse states. These states are comparable to a health index, since they indicate the damage progress and also the fault type.

To estimate the current damage state, and with that also the current status of the component, the different trained HsMMs are applied. Besides the damage state, those models, which most accurately represent the current state sequence, are also

identified. Both the current state and the most probable model form the basis for the subsequent prognosis step. The necessary steps are introduced in the next sections and the algorithm is summarized in Section A.4.

3.4.1 State estimation

The aim of the state estimation step is to find the most adequate damage state based on the trained HsMMs and the current results of the classification by KNN. Besides the damage class C_t, the class probability $\boldsymbol{S}_C(t)$ is also an output of the KNN algorithm, so that the classification uncertainty is considered during the damage state estimation.

The HsMM algorithm, which was introduced in the previous section and in more detail in Section A.2, offers an opportunity to use the same formulae to find the current damage state. Therefore, only the Forward step and parts of the Backward step are necessary to calculate the most probable state sequence for each model. Some extensions are required due to the inspection of $\boldsymbol{S}_C(t)$ instead of the one-dimensional cluster signal C_t. The extensions are introduced in detail in Section A.4. The output of the state estimation is a vector $\boldsymbol{\gamma}_t$, which contains the probabilities of damage states S_i $(i = 1,2,...,N)$ in time step t. Additionally, the duration time d_i inside the current state S_i is counted during the state estimation.

3.4.2 Model identification

In addition to the current state, the most probable HsMM is also identified during the fault diagnosis. Through this, the best fitting damage propagation for the subsequent fault prognosis is selected. The model probability $P(m_j)$ for models m_j $(j = 1,2,...,N)$, each defined with an individual $\boldsymbol{\lambda}$, is calculated with respect to the likelihood $P(\boldsymbol{O}|m_j)$, which is one output of the Forward-Backward algorithm. The model probability is then calculated by:

$$P(m_j) = \frac{P(\boldsymbol{O}|m_j)}{\sum_{k=1}^{N} P(\boldsymbol{O}|m_k)} \tag{3.10}$$

Since several models are running in parallel, a mixture of HsMMs could also most accurately describe the current fault progress. Thus, a mixing algorithm which connects the outputs of the single HsMMs is necessary. One state-of-the-art method is the Interacting Multiple Model (IMM), which is introduced in [Blo84]. Several formulae of this approach are adapted to calculate the mixed model probability

$\mu_k = P(m_{1:N})$. In addition to the opportunity to mix different models, this approach also prevents rapid changes in the model probability, as it can have an imminent low-pass behavior (see Section A.4 for more information).

Another difference to the algorithms of Section 3.3.2 during the training process is that not the entire signal ($t = 1,2,...,T$), but only the latest signal values ($t = T - \Delta T, T - \Delta T + 1, ..., T$) captured with a sliding window of length ΔT are examined. It is assumed that by considering only the current degradation process, the diagnosis accuracy to find the most probable model as well as the actual damage state S_i is increased.

The results of one example of model identification during the complete life cycle of a component is depicted in Figure 3.10. Four runs are applied to train four separate models and as well the net. These models are then compared with a new test run in order to determine the HsMM, which shows a similar behavior to the test run. As depicted, the model of run 2 in particular has a model probability of $P(m_2) \approx 0.82$ over a wide area, whereas models 1 and 4 are completely neglected. Thus, model 2 and the net in particular are considered for the subsequent fault prognosis. Another characteristic of the model identification algorithm is the smooth changes of the net probability around $t = 10\,\mathrm{h}$; these are the result of the algorithm's inherent low-pass behavior.

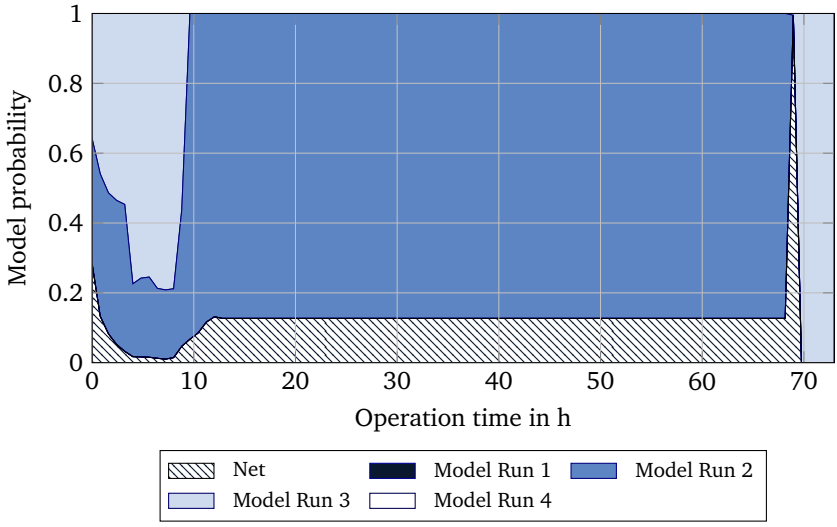


Figure 3.10.: Results of model identification

One implemented feature of the algorithm is that the net probability can be calculated depending on its likelihood as illustrated in Figure 3.10, or manually selected by the user. By selecting a constant value for $P(m_{net})$, the consideration of the net during the prognosis step is guaranteed. The influences of a fixed net probability are evaluated during the verification in Section 3.6.5 and in the validation in Section 5.3.4.

3.5 Prognosis step

Based on the results of the diagnosis, the remaining useful life (RUL) prediction is initiated. Therefore, the trained HsMMs are propagated from the current state S_i until the failure state Ω. The model probability $P(m_j)$ weights the RUL predictions of each HsMM, so that a total RUL prediction is derived. The HsMMs are thereby implemented with the Dynamic Bayesian Network toolbox, presented in [Mur02], which provides a framework to forecast HsMMs.

The first section illustrates, how the duration time within each state is estimated. For this purpose, one duration time estimation method that stems directly from the duration matrix of HsMM is introduced. Another approach comprises the involvement of loads within an interpolation attempt.

To capture the entire stochastic characteristic of each model, a sample approach in the context of Monte-Carlo simulations, also called Markov Chain Monte Carlo (MCMC) methods, is employed. Here, samples run through every relevant model and map their intractable probability distribution. This MCMC approach is introduced in Section 3.5.2. The outcome is an RUL cumulative distribution function (CDF), which is the integrated discrete probability mass function (PMF).

3.5.1 Duration time estimation

Two different approaches to estimating the RUL r_l of a new unit l are examined. In both methods, r_l is calculated by following the state sequence given by the transition matrix $\boldsymbol{A}$ of each HsMM from the current damage state S_i to failure state Ω; simultaneously, the residual duration time $d(S_i)$ within each particular damage state S_i are summed to determine the residual life time. Thus, both methods follow the assumption

$$r_l(i) = -d_*(S_i) + \sum_{k=i}^{\Omega} d(S_k) \tag{3.11}$$

where $d_*(S_i)$ is the already lingered time within state S_i. By the definition of $d(S_i)$, the two approaches are distinct.

The first method stems from the duration matrix $\boldsymbol{D}$ introduced in Section 3.3.2. Examples of entries of $\boldsymbol{D}$ which were derived during the training phase for one particular damage state S_i are plotted in Figure 3.11. The depicted probability density function (PDF) results from a superimposition of three different Gaussian distributions. The RUL is estimated by sampling this PDF using a Monte Carlo approach, which is presented in the next section. The results are then evaluated by means of Equation 3.11.

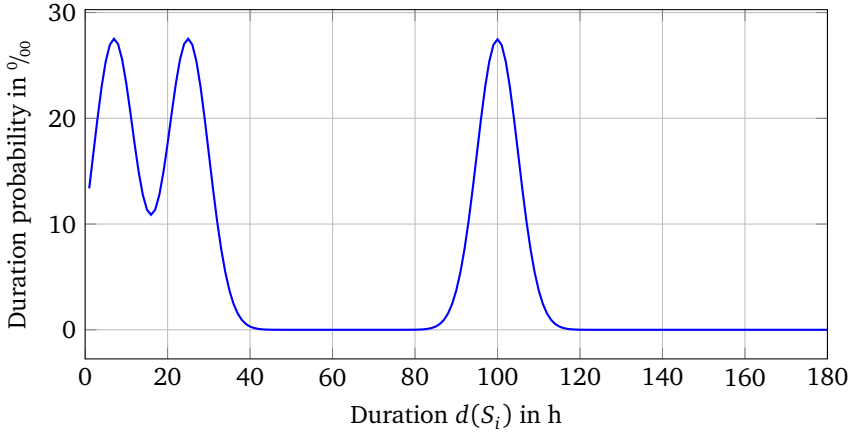


Figure 3.11.: Example of duration probability within damage state S_i

One drawback of this first method is the variety of possible durations within a damage state in the cases such as that shown in Figure 3.11. By adding more information about the current degradation state, it is assumed that the duration time estimation is more accurate. Thus, the second approach uses the already borne load $\boldsymbol{u}_\Sigma$ as another input for the duration time estimation, so that the duration d is a function of $d = f(S_i, u_\Sigma(i))$.

The idea to apply the cumulative load $\boldsymbol{u}_\Sigma$ is based on the Palmgren-Miner Rule, presented for example in [Sch10, pp. 297–299]. This states that every loaded component has a fatigue life endurance, and if the borne load is equal to this endurance, a failure occurs. Some approaches based on this rule are presented in [PJ09] or [NAP10].

By applying this theory to HsMM, the duration time $d(S_i)$ is estimated with respect to the cumulative load u_Σ. One example of the relation between u_Σ and

$d(S_i)$ is depicted in Figure 3.12. Here, the relation between input u_Σ and output $d(S_i)$ is assumed to be hyperbolic, but this differs depending on the degradation process within the damage state.

With regard to prediction, the function $d = f(S_i, u_\Sigma(i))$ is evaluated by an interpolation approach. Thus, $d(S_i)$ and its corresponding $u_\Sigma(S_i)$ are extracted during the HsMM training and form the training points as plotted in Figure 3.12. New durations of unknown $u_\Sigma(S_i)$ are derived by a linear interpolation between these training points; if the new $u_\Sigma(S_i)$ does not lie between the training points or there is only one point for the particular damage state, the last point is simply held. In addition, a Gaussian duration uncertainty around the expected mean is postulated to retain the stochastic character of the first approach in Figure 3.11. Possible benefits arising from this approach are evaluated in Section 5.3.2.

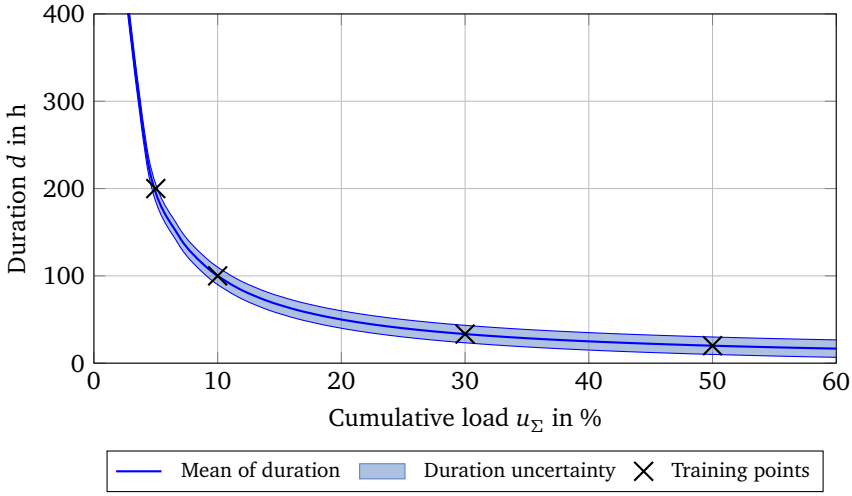


Figure 3.12.: Determination of duration by means of cumulative load u_Σ

Since this is a first trial to include the cumulative load u_Σ into the duration estimation, the applied interpolation method is rather simple and can be easily exchanged by more advanced regression techniques such as Gaussian Process or others.

3.5.2 Markov Chain Monte Carlo approach

The proposed prognostic approach comprises several stochastic effects which result from embedding various probabilities: besides the model probability $P(m_j)$ and the state probability $\boldsymbol{\gamma}$, which are the two outputs of the state estimation, the duration time estimation within a state is also stochastic. Since the resulting duration time probability distribution illustrated in Figure 3.11 for example is not Gaussian, an exact inference is untraceable. Thus, the resulting RUL probability distribution is approximated by a sample approach in the field of Monte Carlo simulations.

The evaluation of Markov models or its extension Hidden semi-Markov Models (HsMM) by Monte Carlo simulations falls into the area of Markov Chain Monte Carlo (MCMC) algorithms. In [Bis13, pp. 537–546], an introduction to this field is given. The RUL probability distribution is approximated by sampling Equation 3.11 from the time of ith prediction to failure state Ω. Thereby, every sample experiences other state transitions and duration times, so that with respect to the model variety of applied HsMM the number of samples n_s is increased to cover all stochastic characteristics. Thus, the result is a discrete PMF for every n_s achieved RUL estimation. This PMF is integrated to a cumulative distribution function (CDF).

The number of samples n_s is determined by calculating the convergence of the RUL CDF. The Two-sample Kolmogorov-Smirnov test offers the opportunity to check whether two stochastic variables are drawn from the same probability distribution. An introduction to the topic of comparing two independent distributions is presented in [Fie15, pp. 217–228]. The convergence of the RUL probability distribution is estimated by comparing two results of MCMC: one with a high number of samples n_s and another with a reduced sample size. As illustrated in Figure 3.13, the CDF with a low sample size of $n_s = 30$ deviates greatly from the smooth CDF with $n_s = 2000$. The number of samples is identified when the null hypothesis that *the two stochastic variables stem from the same probability distribution* of the Two-sample Kolmogorov-Smirnov test at a significance level of $\alpha_{KS} = 0.01$ is not further rejected for an increasing number of samples. The result of this procedure is a sample size of $n_s = 1000$. Since the consequent CDF with $n_s = 1000$ would be congruent to $n_s = 2000$, the plotting in Figure 3.13 for this case is omitted.

Since the resulting RUL distribution is not Gaussian, the 5th, 50th, and 95th percentile are applied for the evaluation of the RUL predictions. The median at 50th quantile is especially in the focus of the estimation error, whereas the 5th and 95th percentiles express the prediction uncertainty. In the case of an exemplary Gaussian distribution, the covered 90 % uncertainty is equal to a $\pm 1.64\,\sigma$ bound.

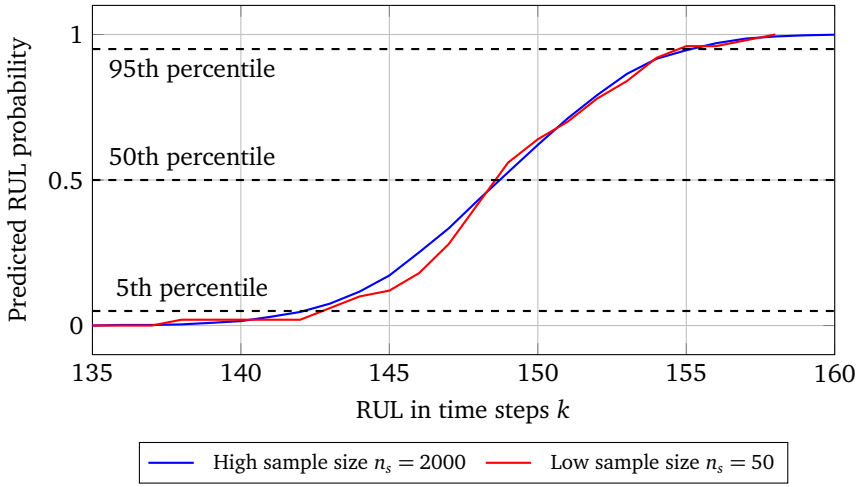


Figure 3.13.: Identical RUL probability distribution with two different sample sizes depicted as CDFs

3.6 Verification of the proposed PHM algorithm

In [DV07], the verification of PHM technology is defined as the process in order to determine whether the technology accurately matches the developer's predefined design specifications. In comparison to the validation, it is an internal process which includes dynamic testing of the algorithm's properties [Jen13, pp. 173–174].

In the field of computer science, both formal and informal verifications are a class of widespread techniques to prove the correctness of algorithms. The formal verification automatically tests, whether all mathematical models of the system function correctly on given inputs [Har03]. Informal verification relies on qualitative instead of quantitative examination of the algorithm. Walkthroughs are an example of informal verification methods [SB10, pp. 340–345]. Parts of both approaches are adapted in order to test the proposed PHM algorithm. Since the output of an unknown degradation test case is hard to predict, verification is provided by testing the algorithm with one of the training datasets. Thus, the output of every step is predictable and it is possible to test, whether the predefined specifications such as prediction accuracy are fulfilled.

The data for the verification stem from the application introduced in Chapter 4 and mentioned in the preamble of this chapter. Thus, the data is an outcome of a

test rig designed to investigate bearing damage in an induction machine. However, similar data are expected in the case of other degrading rotary components such as gear boxes. Thus, the data is one example of a large variety of possible applications.

The sections of the verification follow the structure of Figure 3.1 and Figure 3.2. At the end of the verification in Section 3.6.5, the focus lies on the net approach, as introduced in Section 2.6.

3.6.1 Signal processing

The features are first transformed into the frequency domain and are then selected in accordance with the mutual information criterion, as presented in Section 3.2. To verify this step, three different features of similar frequency ranges are plotted over the entire life cycle of the component in Figure 3.14. Since all features comprise the same information and thus have a high value of mutual information, two of them are removed from the feature subset. The remaining features form the feature subset $\mathbf{\Phi}$, which is transferred to the model training. One example of this feature subset for two different training runs is given in Section A.6. The number of features varies between 3 and 5 with respect to the number of training datasets. Since the probability of differences in features increases with a higher number of training sets, more features are required. Although this cannot be validated within this thesis due to lack of training data, it is assumed that for an increasing amount of training data the quantity of necessary features will converge. Otherwise, the curse of dimensionality would lead to inaccuracies in fault diagnosis and prognosis.

3.6.2 Model training

The feature subset $\mathbf{\Phi}$ of the previous section is handed over to the model training step of Section 3.3 where KNN and the HsMMs are trained. The output of this training is plotted in Figure 3.15 for four training runs. Based on Sil and BIC, both introduced in Section 3.3.3, the optimal number of damage classes M and damage states N, respectively, are identified. Values of BIC during this verification are presented in Section A.6.

Similarly to Figure 3.9, the results of Figure 3.15 show a nearly monotonic rising behavior of the damage states in each run. Only two damage states in run 1 and one damage state in run 4 are entered twice during the entire life cycle. Damage state 1 is also entered in all runs, which indicates a comparable nominal condition at the beginning of each life cycle. In addition, damage states 8 and 9 are selected in runs 1 and 2 as well as in runs 2 and 4, respectively; all of these state entries occur near the components' end of life. This confirms the proposed holistic approach,

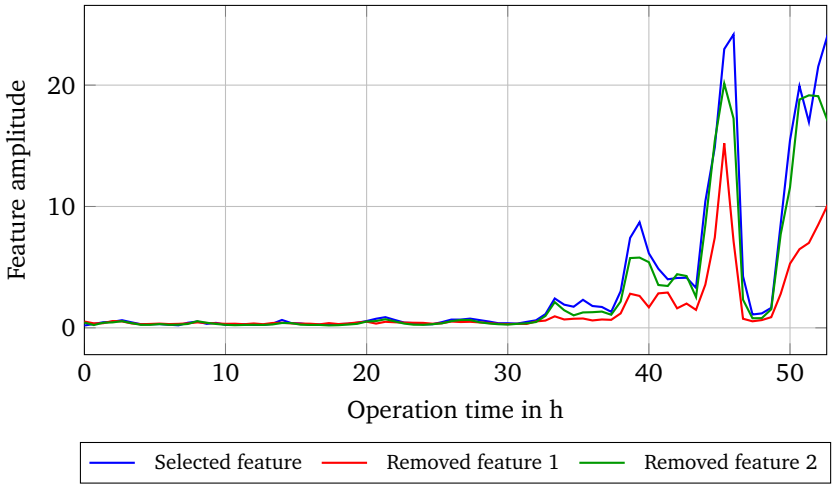


Figure 3.14.: Three features with high mutual information

since several components run through a comparable degradation state. Thus, the generated models are capable of fault diagnosis and prognosis.

3.6.3 Fault diagnosis

Based on the feature subset $\mathbf{\Phi}$ of Section 3.6.1 and the trained KNN and HsMMs of Section 3.6.2, the fault diagnosis is initiated. For this, $\mathbf{\Phi}$ is classified by KNN in the context of the proposed verification method, where the test data is equal to one of the training datasets. In Figure 3.16 the results of this classification are plotted. Here, the round marks represent the training points which are the basis for KNN training, the color distinguishes the resulting classes, and the cross marks are the example test points. Although actually the discrete feature matrix $\mathbf{\Phi}$ of Figure 3.5 is the input for the classification, for a better visualization the continuous features $\mathbf{\Phi}_c$ (plotted as an example in Figure 3.4) are applied. Otherwise, all training points would cover each other and no distinction between the different classes would be possible. The result of this verification test is that they are classified without any misclassification, since the test points of $\mathbf{\Phi}$ overlap the training points of KNN. Thus, the classification score S_C also equals one for every test point.

These results are transferred to the model identification and damage state estimation step. By applying the Forward-Backward algorithm introduced in Section 3.4.1 and Section 3.4.2, the most probable model as well as the best fitting

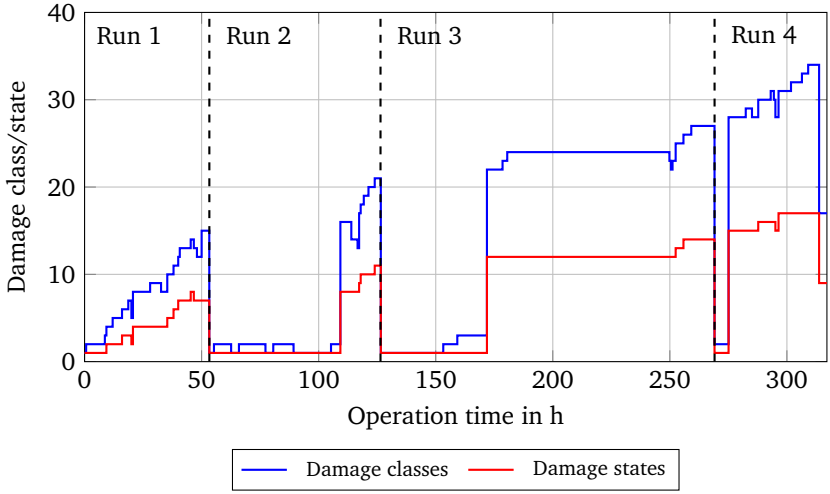


Figure 3.15.: Training of HsMMs with four different runs

state sequence is identified in Figure 3.17. Since the true model and state sequence is known in the context of verification, the outcome of both steps is predictable.

The results of model identification are plotted in Figure 3.17a, where the tested run and the training run of model 2 are the same. As expected, the algorithm identifies the true model 2 after a short period of time ($t \approx 4\,\mathrm{h}$) as being most suitable. The fact that the model probability of model 2 in the first four hours is almost null and simultaneously the probability of model 3 is high can be explained by connecting the model probability with the emission matrix $\boldsymbol{B}$ of the trained HsMMs of Figure 3.15. All runs start at damage state S_1, which is connected to the damage classes 1 to 3. As introduced in Section 3.3.2, the emission matrix $\boldsymbol{B}$ of Equation 3.5 links the damage states with the damage classes. The emission matrices of each HsMM can be also interpreted as the ratio of each damage class within a particular damage state. In the verification example at hand, the emission matrix entries for the damage state $q_t = 1$ and the damage classes $v_k = [1, 2, 3]$ are $\boldsymbol{b}^2_{1,1:3} = P(v_{1:3}|q_t = 1) = [0.43, 0.57, 0]$ for model 2 and $\boldsymbol{b}^3_{1,1:3} = [0.59, 0.13, 0.28]$ for model 3. The state estimation for the test dataset (equal to model 2) starts with a damage class $v = 1$ during the first four hours and thus, the first entries $\boldsymbol{b}^2_{1,1}$ and $\boldsymbol{b}^3_{1,1}$ in particular are the reason for the deviations in model probability. Since $\boldsymbol{b}^3_{1,1}$ is higher than $\boldsymbol{b}^2_{1,1}$, consequently the probability of model 3 being the most suitable

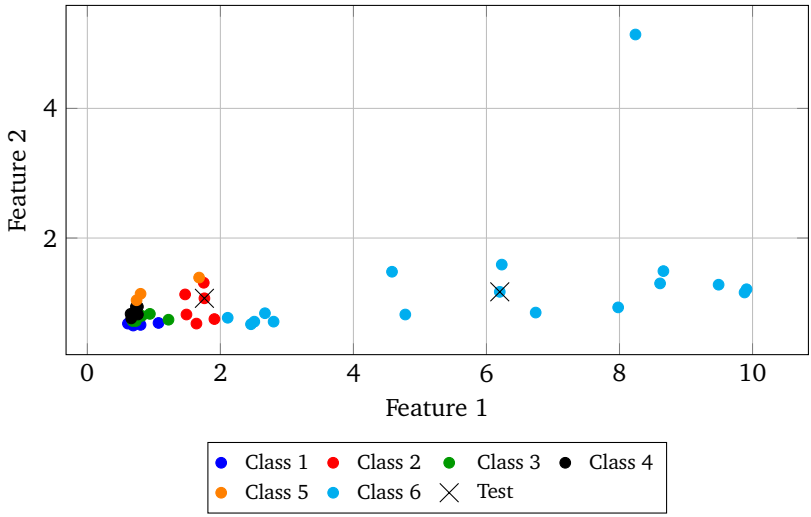


Figure 3.16.: KNN classification during verification

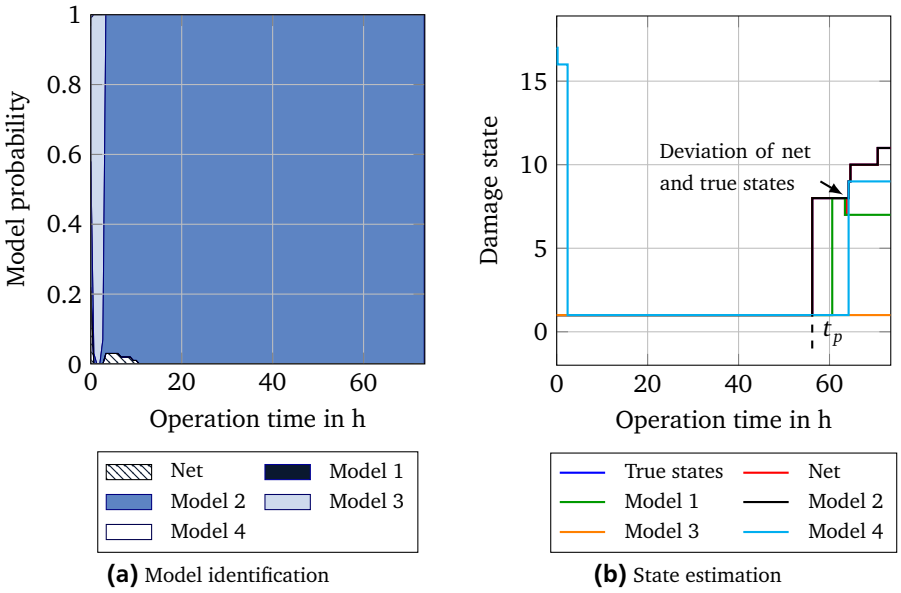
is more likely, in accordance to Equation 3.10. After the first four hours, damage class $\nu = 2$ is detected, so that the model probability for model 2 tends to 1.

The state estimations plotted in Figure 3.17b show that models 1, 3, and 4 have difficulties finding the true states. This is due to the fact that the actual damage states of model 2 are not included in these models with the exception of damage state S_1. In contrast, the true states and the estimated states of model 2 are equal throughout the life span, which shows that the state estimation works as expected. In addition, the net is also able to predict the true states with only one small deviation at $t \approx 65\,\mathrm{h}$. The time of the first prediction t_p is also marked in this figure: when the state transition from nominal state S_1 to S_8 is executed, a fault is detected.

3.6.4 Fault prognosis

With the model probability and state estimation of Figure 3.17 the fault prognosis of the different trained HsMMs is initiated at time t_p. Here, the state estimation provides the initial state of the HsMM propagation, whereas based on the model probability, the sampled RUL prediction probabilities of each particular HsMM are combined.

The first example addresses the prediction of model 2, which is also the applied example of Section 3.6.3. Since the damage state sequence of this model presented



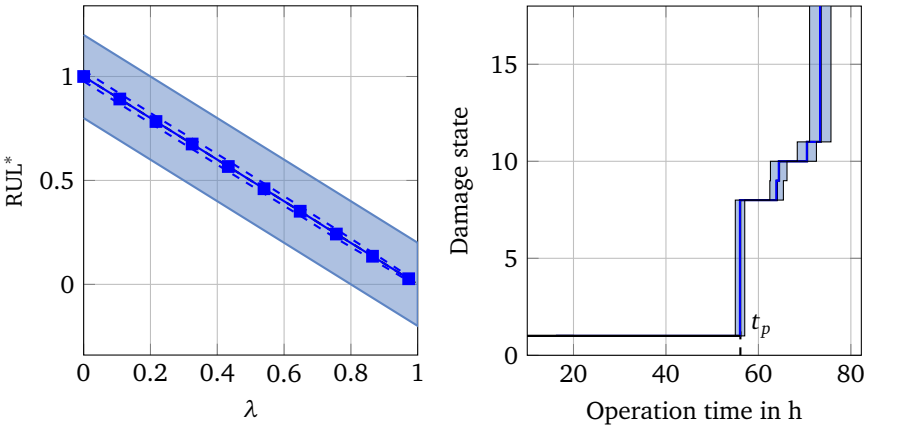
(a) Model identification **(b)** State estimation

Figure 3.17.: Model identification and state estimation for four training datasets (the tested dataset is also included in the training data of model 2)

in Figure 3.15 has no internal cycles (i.e. no damage state is entered multiple times) the verification by means of this dataset is expected to show ideal results. In Figure 3.18 the RUL predictions for this case are plotted. Figure 3.18a represents the typical plot to determine the prognostic horizon (PH), that was introduced in Section 2.4, where the RUL predictions are normalized with the total prediction span t_{PC} of the unit under test. The predicted RUL lies within the predetermined accuracy bound of $\alpha = 0.2$ throughout the entire life cycle.

Besides the exact match of predicted median and true RUL, the RUL uncertainty is also small. This is because all samples follow a similar path through the HsMM state transitions. In Figure 3.18b, this path is plotted from the prediction time $t = t_p = 56\,\mathrm{h}$ ($\lambda = 0$) to the point of failure state $\Omega = 18$ ($\lambda = 1$). Besides the true states, the median and the prediction uncertainty between 5th and 95th percentile of the resulting sampled RUL CDF is also plotted. Due to the high prediction accuracy during this verification example, the true state sequence is covered by the predicted states throughout the prediction. The plotted uncertainty bound slightly increases after every exit of a damage state. The reason for this is the applied

duration model of Figure 3.11. However, instead of three superposed Gaussian distributions, only one distribution is applied in Figure 3.18b. During each state entry, the duration time within the state is calculated based on a Gaussian distribution. Thus, the increasing uncertainty is a result of cascaded Gaussian distributions, where the current covariance is simply added to the previous covariance.



(a) Ideal RUL prediction

(b) State prediction at $t = 56\,\mathrm{h}$

Figure 3.18.: Fault prognosis during verification (the tested dataset is also included in the training data of model 2)

As expected, these results characterize a benchmark for other investigations concerning RUL predictions. The trained model in combination with the applied verification concept (training data is used for testing) provided ideal results, which shows the mathematical correctness of the implementation. In Figure 3.19, another set of RUL predictions with the dataset of model 4 is presented in the context of the same verification concept. However, this dataset differs from the dataset of model 2, since the corresponding HsMM consists of a cycle in state transition (damage state S_{15} is entered twice). The results are depicted in Figure 3.19. In Figure 3.19a, the RUL predictions of this dataset over the normed life cycle are plotted. This figure also exhibits differences in the two introduced approaches for

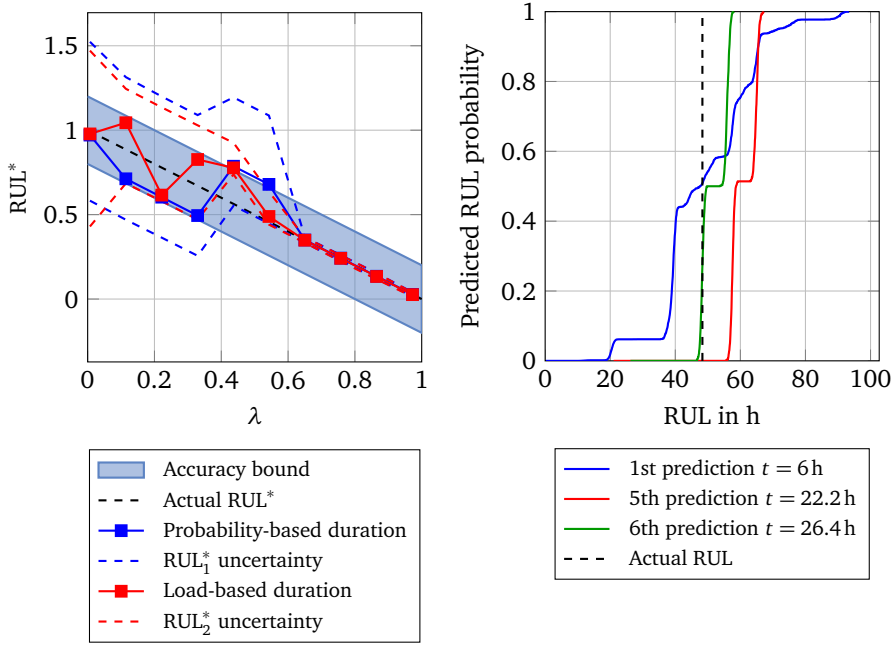
the calculation of the duration time within a state. As mentioned in Section 3.5.1, both the probability-based and the load-based methods were investigated. Both approaches show similar deviations of predicted and actual RUL. The effects of the state transition cycle are visible especially for the probability-based approach: until $\lambda \approx 0.35$, the predictions tend to underestimate the actual RUL (false positive), since some samples omit parts of the trained state path. By examining the corresponding state sequence of run 4 in Figure 3.15, it means that starting in damage state S_{15}, the subsequent damage state S_{16} is skipped and S_{17} is directly entered. The 5th and 6th predictions around $\lambda = 0.5$ drift to a false negative prediction error, which is caused by re-entering S_{15} multiple times. In both methods the predictions in the last third of the component life cycle show a comparable behavior concerning prediction error and RUL uncertainty, as in Figure 3.18a. The comparison of both duration estimation methods indicates slight benefits of the load-based approach concerning prediction error. This results in differences in prognostic horizon ($\mathrm{PH}_1 = 0.35\,\lambda$, $\mathrm{PH}_2 = 1\,\lambda$) or in mean absolute percentage error ($\mathrm{MAPE}_1 = 15\,\%$, $\mathrm{MAPE}_2 = 13\,\%$).

In addition, three different prediction probabilities are plotted in Figure 3.19b over estimated RUL, which are the basis for the previously presented predictions. This demonstrates, how the RUL predictions converge while the component's life cycle proceeds. The true length of the life cycle is also marked. These CDFs correspond to the results of the load-based method of Figure 3.19a. The first prediction shows a rather broad probability spread from 20 h (5th percentile), over 48 h (median) to 78 h (95th percentile). The CDFs of the 5th and 6th prediction are considerably narrower, although the 5th prediction has an offset.

The contrast between the results of Figure 3.18a and Figure 3.19a show the need to restrain the occurrence of transition cycles. However, due to the underlying training data these cycles cannot be completely excluded. A limited number of features, a small resolution during the discretization of the features, or the optimized number of damage states and classes are reasons for this. Thus, the number of damage state entries are constrained (see Section 3.3.4).

3.6.5 Net approach

In Section 3.4.2, the possibility of manually selecting a constant net probability $P(m_{net})$ was introduced. The influences of this approach during the verification with the data of model 4 are presented in Figure 3.20. In Figure 3.20a the model probability in the case of a net probability of 30 % is plotted. Similarly to Figure 3.17a, mainly the model which is trained with the test data is selected to be the



(a) Comparison of the two approaches for duration time estimation (see Section 3.5.1)

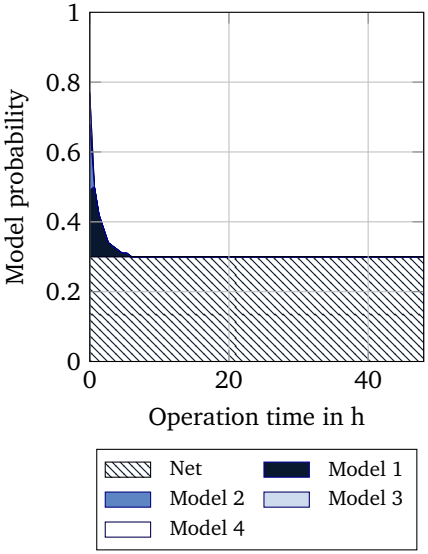
(b) Three RUL prediction probabilities (load-based HsMM)

Figure 3.19.: Fault prognosis during verification (the tested dataset is also included in the training data of model 4)

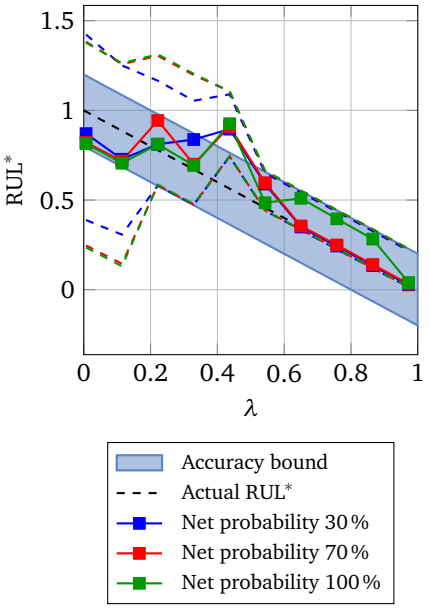
most suitable. Additionally, the model probability of the net remains $P(m_{net}) = 0.3$ as demanded.

In Figure 3.20b the impact on the fault prediction with three different net probabilities (30 %, 70 %, 100 %) is presented. The load-based duration estimation is selected for the analysis, which can be compared to the corresponding results of Figure 3.19a for a likelihood-dependent net probability ($P(m_{net}) < 0.01$). Only slight differences between the three approaches are identified. Merely at the end of the component's life cycle does the algorithm with a net probability of 100 % produce a deviation of the actual RUL, whereas the prediction error of the other two approaches are null. In comparison to the approach of Figure 3.19a, the prediction uncertainty is increased and the PH is smaller, since all algorithms of Figure 3.20a leave the accuracy bound during the 5th prediction.

Thus, the manual selection of a net probability is not beneficial in the context of verification. In the case of unknown test data, this is expected to increase the robustness of the prediction, since all potential state transitions are considered. This is therefore examined in detail in Section 5.3.4.



(a) Model identification (net probability 30 %)



(b) Differrent net probabilities (corresponding prediction uncertainties in dashed lines)

Figure 3.20.: Model identification and load-based fault prediction during verification with varying net probabilities (the tested dataset is also included in the training data of model 4)

3.7 Comparative algorithm

The abovementioned PHM algorithms are compared to state-of-the-art approaches during the validation in Section 5.3.3 and Section 5.4.2. Therefore, parts of the PHM algorithm introduced in [MTMZ12] and [TMMZT12] are adapted and included in the postulated framework of this thesis. Instead of replacing the entire algorithm, only the basic idea behind the model building of the HsMMs is

exchanged in order to rather test the two different concepts instead of evaluating one particular cluster or classification algorithm. Thus, the original concept is presented and the implementation within this thesis is then introduced.

The idea behind the concept presented in [MTMZ12] and [TMMZT12] is to treat every training dataset of each degraded component individually. Thus, instead of finding holistic damage states in different training datasets, each run generates independent damage states. Therefore, the number of damage states N of the individual HsMMs is set to $N = 3$ in order to distinguish the damage states "nominal", "initiation and propagation of defect", and "failure". These states are expected to be entered several times during a degradation process, so that the number of state entries is counted during the training process. Later, in the context of fault prognosis, the number of state entries is constrained. The stay durations of each state are then assumed to follow a Gaussian distribution or a mixture distribution of several normal distributions, similarly to the proposed probability-based approach of Section 3.5.1. This concept for model training is adapted for a comparative algorithm.

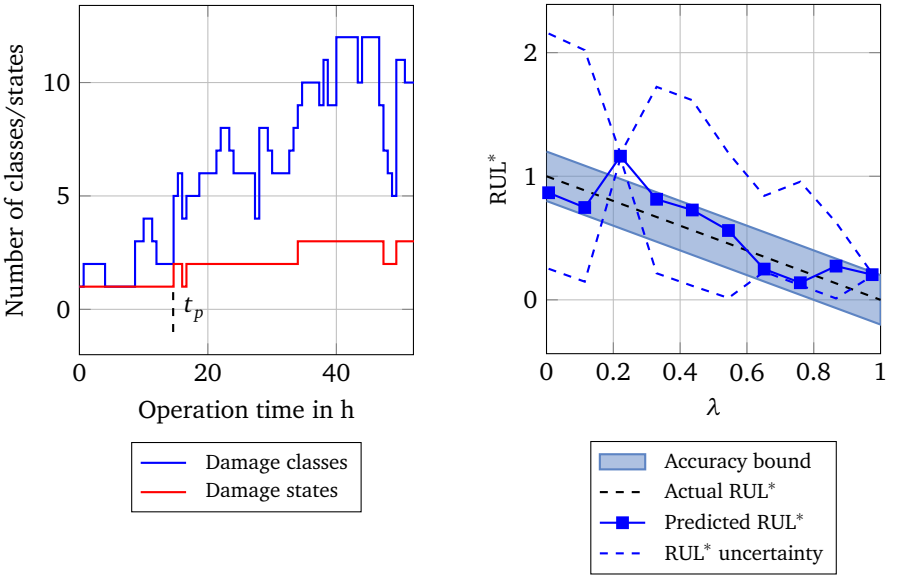
With regard to the underlying features, in [MTMZ12] and [TMMZT12] a multi-feature approach (similar to the proposed PHM algorithm) is used. The features are clustered by the mixture of Gaussian algorithm. The output of this clustering is directly used as the emission matrix $\boldsymbol{B}$ of the HsMM. The number of clusters M is determined by training HsMMs with a varying M and examining the behavior of the resulting likelihood, which is calculated during the Forward-Backward algorithm. Since the likelihood is expected to converge for an increasing number of M, the value M at the beginning of this convergence is selected. In [MTMZ12] this value is $M = 4$. To reduce the differences between the proposed approach of this thesis and the comparative algorithm, the feature generation and selection of the proposed concept is applied. Additionally, the K-means for clustering and KNN for classification are chosen, as introduced in Section 3.3.1, but individually for every training dataset.

After the training phase of each HsMM, a test dataset can be examined. The current damage state is determined by evaluating each particular HsMM with the current set of features. The output of this state estimation may differ among the trained HsMMs, i.e. the estimated damage state for model 1 is different to the expected state of another HsMM. Simultaneously, the model probability of each HsMM is calculated. Only the model with the highest model probability is selected for the prediction step. After the transition from nominal state to fault state is detected by the state estimation (change from S_1^i to S_2^i in model m_i), the prediction begins in t_p. The RUL is estimated by identifying the longest and shortest path through the damage states. Based on the Gaussian distributed duration times

within each state, the RUL is estimated by summing over all durations, as introduced in Equation 3.11. This procedure is also exchanged by the MCMC approach of Section 3.5.2.

Some examples of results of this comparative algorithm with the discussed changes to the original approach of [MTMZ12] and [TMMZT12] are illustrated in Figure 3.21. Besides the outcome of the clustering and HsMM training in Figure 3.21a, one set of predictions in the context of verification is also given in Figure 3.21b. Analogous to the verification in Section 3.6.4, the tested dataset is again included in the training data. The damage states of the trained model in Figure 3.21a show a similar behavior to that presented in Figure 12 of [MTMZ12]. However, the number of entries in each damage state is lower in comparison to the original approach. However, in the context of an internal model validity, the three damage states "nominal", "initiation and propagation of defect", and "failure" are entered in a reasonable sequence.

The fault prediction in Figure 3.21b is comparable to the prediction performance of the proposed algorithm. The prediction errors stem mainly from the probability-dependent duration estimation, since the differences in durations within damage state S_2 and S_3 differ greatly. In addition, the outlier at $\lambda \approx 0.2$ follows from a falsely selected model during the model identification. The prediction uncertainty is also higher, which is a result of the approach-dependent internal cycles.



(a) Trained model of comparative algorithm

(b) Fault prediction during verification (tested dataset is also included in training data)

Figure 3.21.: Model training and prediction in the case of comparative algorithm during verification

4 Bearing damage as case study for the proposed PHM algorithm

The previously defined PHM algorithm is evaluated by data of degrading bearings, as described in Section 2.2.2. Thus, this chapter is divided into three parts. First, a brief overview of currently established methods for bearing degradation and the analyzed signals in terms of condition monitoring (CM) is given in Section 4.1. Based on this overview, requirements are derived, which are the foundation for the design of a new test rig (Section 4.2). At the end of this chapter, the final test rig setup that was applied for the data generation is presented in Section 4.3.

4.1 Trends of PHM for bearing degradation

This section provides an introduction to state-of-the-art methods for generating and examining bearing degradation data. Thus, a small overview of research trends related to the examined signal types of bearing fault is given in Section 4.1.1. Realistic damage and failure during the entire life cycle of a bearing are introduced in Section 4.1.2. The goal of a test rig is to reproduce similar damage processes in a short time. Therefore, the established methods for accelerated bearing aging are discussed in the subsequent section.

4.1.1 Examined signal types

Analogous to the examined signals in rotary machines in Section 2.3.1, the focus for the detection of bearing failures in terms of CM is on signals of vibration, acoustic emission, debris, temperature, and position [TSS06]. In particular, the effects of bearing faults on the vibration signal have been explored excessively; from modeling ([MS85], [Ant07], [YGA11]) to diagnosis ([BBV14], [Nad04], [OLD07]) and to prognosis ([LBC15], [MTMZ12], [JOL08]) the vibration signal is the state-of-the-art signal to validate new approaches.

Additionally, the motor current signature analysis (MCSA), i.e. the analysis of motor current phases, with regard to bearing fault diagnosis is the focus of many publications. The causes of the effects of bearing faults on motor current are radial

movement of the rotor center on one hand and load torque variations on the other [BGRR08]. As depicted in Figure 4.1, due to the angular deflection $\varphi(t)$ of the rotor, the magnetic flux density varies, since the air gap $l_{air}(t)$ between rotor and stator oscillates. Thus, a voltage is induced, which is superposed to the sinusoidal phase voltage of the induction machine. This imperfection is measurable in the context of MCSA. As described and motivated in Section 2.3.1 and Section 2.6, the MCSA is applied in order to validate the defined PHM algorithm. Thus, the focus of the test rig is the investigation of bearing faults in induction machines.

Since the MCSA is mainly carried out in the frequency domain, the fault frequencies are of interest. Several publications base the expected frequency range of the fault frequencies upon the equations of Palmgren as introduced in Section 2.3.2.3. Since these equations are only valid for fault frequencies in vibration signals, [SHKB95] set up a relation between the Palmgren equations and the fault frequencies in the motor current. They assume that the fault frequencies f_v in the vibration signal are detectable in the stator current with

$$f_s = |f_e \pm m \cdot f_v| \tag{4.1}$$

where $m = 1, 2, 3, ..., n$ describes the order of harmonics and f_e is the supply frequency of the induction machine. In accordance with these equations [Bel08], [ZP07], and [RRBM02] identify the effects of bearing faults in the range of 30 Hz to 150 Hz. Others ([ICBR09], [JA05], [SHH04b]) investigate a range of up to 800 Hz so that this limit must be considered for the design of the test rig's sensor suite.

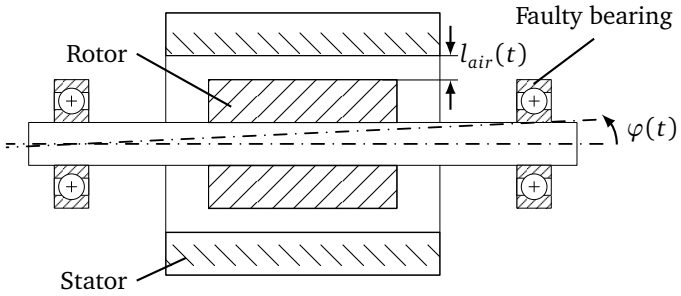


Figure 4.1.: Effects of bearing faults on the air gap between rotor and stator

4.1.2 Common bearing damages

In accordance with [SHH04a], bearing faults are categorized into *single point defects* and *generalized roughness*. Single point defects cause the abovementioned characteristic frequencies, since the deflection of the rotor in stationary state is idealized deterministic. An example of single point defects are spalls or pits on the bearing components. Generalized roughness includes bearings that are rough, irregular, or deformed over a large area. Thus, the equations by Palmgren are not valid for this type of fault, since the effects are highly stochastic and stimulate a broader frequency area. The causes of general roughness are contamination or loss of lubricant, shaft currents, misalignment, or corrosion. The results are scratches or pits over the entire component. One significant damage pattern is the appearance of chatter marks due to bearing currents. [SHH04a]

4.1.3 Methods for bearing degradation

The aim of artificial bearing degradation is to reproduce the faults described in the previous section as realistically as possible. According to [SHH05], two different approaches to the generation of bearing faults are widely used for the validation of diagnosis and prognosis methods. They can be classified as *off-line faults*, when the tested bearings are removed from their in situ position, and *on-line faults*, which are generated by means of measurements for the accelerated bearing aging. Off-line faults can be further divided into artificially pre-damaging e.g. by drilling holes into the rings ([ECBC12], [ED04]), scratching the surface of the raceways or rolling elements [ICBR09], and faults that occur during real operations. [BGRR08] verify and validate their bearing fault models by using of bearings which had been issued from industrial maintenance.

In [SHH05] the problems of both off-line approaches are discussed. They state that since the changes to the probed machine during the removal and mounting of the bearings lead to differing test conditions, the experimental data is corrupted. Thus, both methods are mainly used to generate data for fault diagnosis. To avoid these problems, the on-line approach of producing an artificially accelerated bearing damage is beneficial.

Several methods for accelerated bearing aging are presented in the literature. All approaches have in common the feature that the bearings are stressed with a load that diverges from the approved design point of the bearing. One approach is mechanical forces (radial [MS85], axial [JOL08]) and torques ([KP02], [RRBM02]) to generate a systematic degradation. [JOL08] combine the radial force with a high operating temperature (about 127 °C) to further reduce the overall lifespan of new

bearings to approximately one month. The artificial contamination of the lubricant with dust [KCT+14] or alumina particles [ANMK06] is another method to achieve a mainly monotonic degradation behavior.

One further approach to generating on-line bearing failure is the application of a current that flows directly through the test bearing. This method is known as bearing current. The realistic correspondent is discussed in [BH02] in the case of paper machines. Here, the control unit of the paper machine created a potential on the shaft, which led to electrical discharge machining (EDM) of the bearing. The EDM produced chatter marks on the inner bearing surface of all examined motors so that it was necessary for the bearings to be exchanged.

Besides the existence of a potential, the appearance of EDM also depends on the state of lubrication, since the oil film around rolling elements has an insulating effect. When the oil film is thin enough (for example in areas of high load) spark erosions arise, which generate small pits burned into the races. [SHH05] use this approach to generate on-line failures within a range from several hours to three weeks, dependent on the amount of lubricant and the applied voltage of the bearing current (sine with a maximum voltage of 5 V at 60 Hz).

4.2 Design of a test rig for accelerated bearing aging

In the previous section, established methods of bearing degradation were introduced. Thus, in this section the derivative requirements for a new test rig in alignment with the delimitations to other approaches of Section 2.6 are presented. The final concept is outlined in Section 4.2.2.

4.2.1 Derivative requirements

The derivative requirements of the test rig are summarized in Table 4.1. A couple of requirements arise, since some suitable components of former test rigs were already available. These are the type of motor, the power supply, or the data acquisition system. Thus, the inclusion of these parts into the design is mandatory. Other requirements such as the type of load, the examined bearings, the measured signals, or the sample rate of the sensor suite are derived from the experiences of Section 4.1. The latter is calculated on the basis of the maximum determined frequency of bearing faults (800 Hz, [SHH04b]) multiplied by factor ten to avoid sampling effects. The requirements for daily operations such as safety issues, easy access, and fast assembly and disassembly of the bearings are summarized in "other requirements".

Table 4.1.: Derivative requirements for the test rig

Motor	
Type	Induction machine by Emod Motoren, type *80S/2*
Supply frequency	50 Hz
Power supply	Elgar type *1001SL*
Bearings	
Type	Single-row ball bearings
Price	Low-priced (maximum 10€ per part)
Loads	
Bearing current	Minimum voltage of 5 V, supply frequency of 50 Hz
Radial load	Above design point of the bearing
Contamination	Access to inside of bearing possible
Sensors	
Sample rate	Minimum of 8 kHz
Stator current	All three current phases, low signal to noise ratio
Vibration	Close to bearing, in radial direction of the bearing
Data acquisition	DSpace 1103 controller board
Other requirements	
Access	Easy (rapid application of other load types)
Assembly/disassembly	Easy (rapid exchange of test bearings)
Safety	Protection against worn parts, bearing current

4.2.2 Final concept of test rig

The previously defined requirements are the basis for a test rig design, which is the focus of this section. A general overview of the individual components of the test rig is provided in Section 4.2.2.1. The measured signals and the sensor concept, including the selected sensors, are presented in Section 4.2.2.2. The test bearings for the run to failure investigations are then given.

An entire test rig description was published in [APK15], and many parts of this section are based on that publication.

4.2.2.1 General overview and applied loads

The test rig at hand was designed in cooperation with students in the context of two design projects ([ABS+13], [BDGM14]). The result of the concept phase is depicted as a CAD sketch in Figure 4.2.

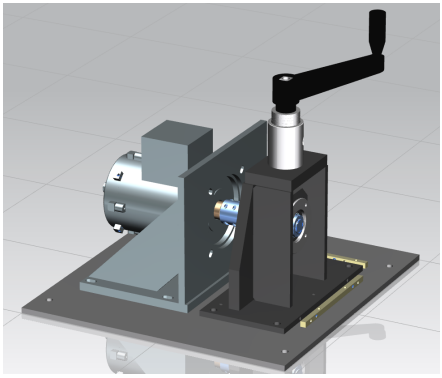


Figure 4.2.: CAD sketch of the test rig with radial force as load

The core element of the test bench is a simple three-phase a.c. motor of type *80S/2*, designed by the manufacturer *Emod Motoren*. The standard power of this motor is 0.75 kW at a line-to-line voltage of 115 V. The supply frequency is $f_e = 50\,\text{Hz}$, it has two terminal pairs and for most of the experiments a slip of about 0.6 %. The motor is depicted on the left of Figure 4.2. The power supply is provided by three type *1001SL* of the manufacturer *Elgar*, which enables a variable supply frequency and voltage.

One difference in comparison to the delivery status of the motor is the displacement of the loose bearing from the casing into a separate bearing bracket, which can be seen on the right side of Figure 4.3. Since this bearing is also the test bearing for the run to failure runs, the disassembly and reset of the test scenario between two runs is more time-efficient. Therefore, a second shaft extends the motor shaft. A stiff coupling, which connects both shafts, ensures that vibrations of the test bearing are directly transmitted to the motor shaft and are therefore, detected by the MCSA.

As required, three different types of load are included in the concept for an artificially accelerated bearing aging: Radial force, bearing current, and contamination of the lubrication. In the case of the radial force configuration as depicted in Figure 4.2, a ball joint bearing that ensures a free angular movement of the shafts

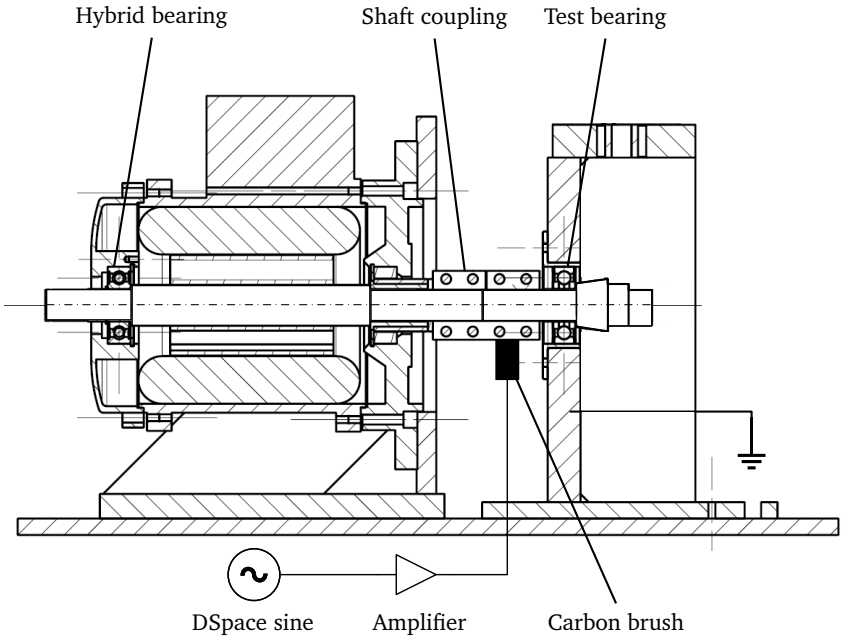


Figure 4.3.: Cross section of the test bench in bearing current configuration

introduces the load. By turning the crank lever, the joint ball bearing is deflected and consequently the test bearing is stressed.

Another change to the motor is caused by the application of bearing current. As it is intended that a current will flow through the test bearing, the motor shaft must be insulated. Thus, the original fixed bearing of the motor is replaced by a hybrid bearing (see Figure 4.3) in order to circumvent a current flow through the motor casing. The current is introduced by a carbon brush which is pressed on the shaft near the test bearing. To imitate the influences of high frequency switching motor drives in accordance with the requirements, the current is rotary with a supply frequency of 50 Hz and a voltage of up to 9 V. A DSpace system provides the voltage signal, which is amplified by an op-amp (*OPA541* by *Burr-Brown*). The circuit is closed by first connecting the bearing bracket with a load resistor, which prevents a real short-circuit. The load resistor is then connected to the ground of the motor, which protects the user from electric shocks.

4.2.2.2 Sensor concept

As required in Section 4.2.1, the data acquisition system is based on a DSpace 1103 controller board and a comprehensive sensor suite, which is listed in Table 4.2. Since one delimitation of this thesis is the evaluation of PHM algorithms by means of stator currents, the phase currents of the motor are measured by three closed loop sensors, which use the Hall Effect for indirect current measurement. Another current sensor measures the bearing current by recording the output of the op-amp. In order to compare and especially to assess the performance of the applied current sensor, an accelerometer is mounted in the radial direction on the bearing bracket next to the test bearing. In addition, the ambient and test bearing temperatures are measured to ensure comparable test procedures and for another assessment of bearing degradation, respectively. The shaft rotation is measured to identify possible changes in shaft movement during the degradation process.

In terms of sample frequency, the vibration and current signals are sampled with a frequency of 25 kHz before the signals are filtered by means of a 10 kHz low-pass filter. Thus, the sample rate is higher than the required sample rate of 8 kHz. The ambient and bearing temperature as well as the rotation speed of the shaft are captured once per second. The recording of measurement data is performed every two minutes. The measure window width is ten seconds and the files are saved in Matlab format *.mat*.

Table 4.2.: Sensor suite of test rig

Sensor	**Type**	**Qty**	**Measurement**
Current sensor	LEM LA 25-NP	3	Motor phase
Current sensor	LEM LA 25-NP	1	Fluting current
Accelerometer	Kistler 8702	1	Radial vibration
Temperature sensor	National Semiconductor LM35	2	Bearing and ambient temperature
Position sensor	Honeywell SS495A1	1	Shaft rotation

4.2.2.3 Test bearings

The test bearing in the bearing bracket is mounted in a hull which can be exchanged. Depending on the diameter D, different types of bearings can be examined. Other differentiating factors are manufacturer, dynamic and static load C

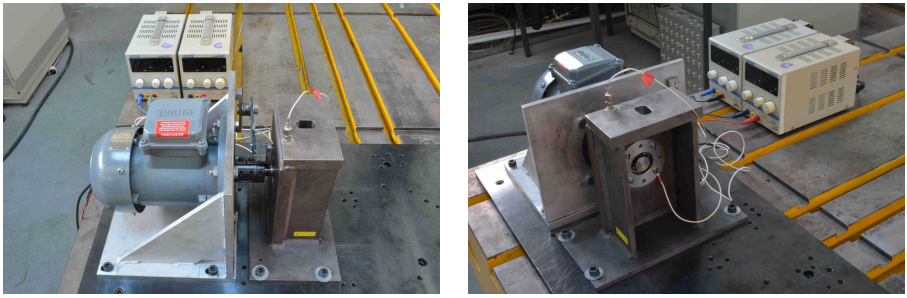
and C_0, the designation and the price per bearing (exclusive VAT). A listing of the employed bearings is given in Table 4.3.

Table 4.3.: Listing of possible bearings

OEM	D [mm]	Rolling element	C/C_0 [kN]	Type No.	Price [€]
SKF	32	Balls	4.0/2.3	61804	8.73
SKF	42	Balls	7.3/4.1	16004	5.81
ISB	32	Balls	4.0/2.3	61804	2.46
ISB	42	Balls	7.1/4.0	16004	2.35
ISB	47	Cylinder	25.0/22.0	NU204	8.30
SKF	42	Balls	10.0/5.0	6004 TN9/C3	7.12

4.3 Test rig setup

The test rig concept of the previous section was assembled in cooperation with a student within a master's thesis [Sch15]. A few changes to the concept arose after first investigations, which are described in this section. The test bench configuration which was used to generate the bearing degradation data for the evaluation of the PHM algorithm is depicted in Figure 4.4.

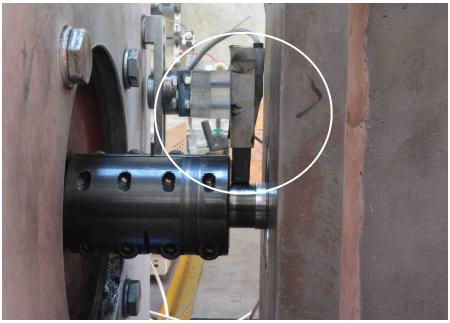


(a) Front **(b)** Right

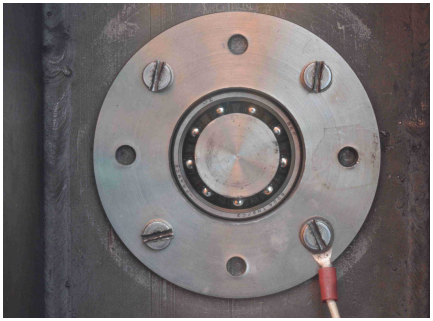
Figure 4.4.: Test rig in the bearing current configuration

With regard to the applied load, only the bearing current was considered, since an overload due to radial force had caused a crack in the motor shaft. As the bear-

ing degradation by EDMs had been sufficient with regard to degradation behavior or timespan of run to failure trials, the application of contaminated lubrication was also not necessary. The implementation of the bearing current is presented in Figure 4.5. The carbon brush is pressed onto the shaft in order to guarantee a permanent current flow. The two power supplies are depicted in Figure 4.4b in the upper right corner. They provide the voltage for the op-amp for negative and positive voltage. The mounted ground wire beside the test bearing is presented in Figure 4.5b.



(a) Carbon brush (white circle)



(b) Ground wiring

Figure 4.5.: Elements for the application of bearing current

The op-amp and the voltage signal are designed to generate a resulting current through the bearing with a maximum amplitude of about 3 A; a cutout of this signal is plotted in Figure 4.6. It can be seen that the real current shows a mismatch in comparison to an ideal sine especially in the voltage area, near null. It is assumed that this behavior is caused by the lubrication, which varies the overall resistance between the outer and inner ring. The lubrication increases the overall resistance so that the current flow is blocked, especially in areas of low voltage.

During first run to failure experiments using bearings with a steel cage, these cages led to an abrupt failure of the tested bearing, which is depicted in Figure 4.7. Thus, there was no significant degradation either in vibration or in current data, which subsequently led to the selection of bearings with polyamide cages (fiberglass reinforced nylon) in order to prevent these sudden failures. The dark polyamide cage can be seen in Figure 4.5b. Therefore, all run to failure runs applied in this thesis were executed with bearings of type SKF 6004 TN9/C3. Here, TN9 stands for the polyamide cage and C3 characterizes increased clearance.

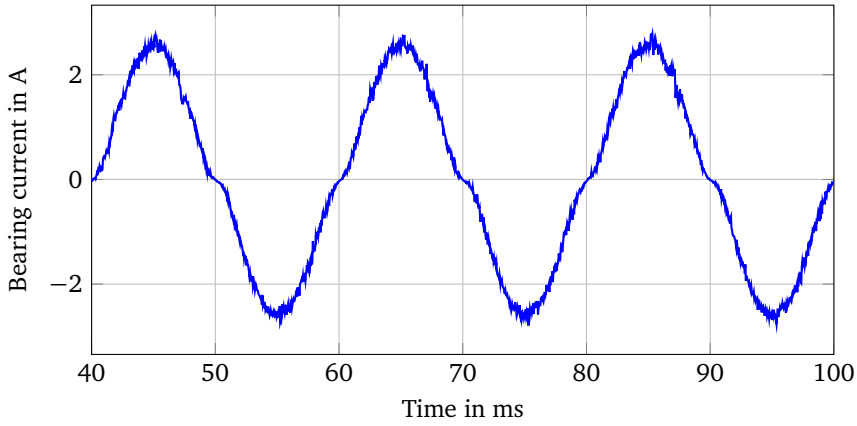


Figure 4.6.: Current through test bearing

Figure 4.7.: Bearing failure due to cage damage

5 Analysis with bearing degradation data

The proposed PHM algorithm is tested in this chapter. For this validation, bearing degradation data is applied. The data was generated by the test rig introduced in the previous chapter. Thus, the first section presents the applied test procedure and provides an overview of the available data. Validation of the fault diagnosis and prognosis steps is split into two parts: first, a validation of both steps with a representative subset of available datasets is executed in the context of a leave one out cross validation (LOOCV) procedure. Besides the prognostic performance of all three Hidden semi-Markov Model (HsMM) methods, the influences of different parameter settings are also analyzed in Section 5.3. Here, especially the input parameters introduced in Section A.5 are applied. The second part comprises a final validation with all datasets in Section 5.4. This enables the assessment of the of the robustness of the algorithms with regard to varying number of training datasets. The final section begins with a general discussion of the achieved prognostic performances of each HsMM approach. Based on the validation results, cost benefits are estimated at the end of this chapter.

5.1 Description of the applied data

In [Jen13, p. 85] the challenge of data availability is discussed. Different data sources and their pros and cons are introduced there. The range of methods reaches from *simulation models*, over *subsystem/system fault characterization tests* to *technology maturation field program*. Each method is assessed with regard to its applicability and realism. The data generation of this thesis is located in the field of system/subsystem fault characterization tests, since the applied induction machine is a complete system or a line-replaceable unit of a superior system. The benefit of this data generation method is *realistic data*, whereas one drawback can be that *the seeded faults are not natural*. [Jen13, p. 85]

This section provides an overview of the applied data, beginning with the selected test procedure for the generation of degradation data. It also presents the selected performance metrics for the subsequent validations. A general overview

of the available datasets is then provided. The achieved damage cases as a consequence of the accelerated bearing aging with bearing current, are illustrated in Section 5.1.3. This section ends with an insight into the available data.

5.1.1 Test procedures

The test rig presented in Section 4.2.2 was utilized in order to generate degradation data of bearings (type *SKF 6004 TN9/C3*) loaded with bearing current. These bearings were in delivery status at the beginning of the experiments, so that no preprocessing steps such as removal or contamination of lubrication as well as initial damage of components was applied. The induction machine was driven under the conditions described in Section 4.2.2.1, with a line voltage of 115 V and a supply frequency of 50 Hz. Except for the bearing current, no further load was applied, so that the rotational speed was almost equal to the supply frequency with only a slight slip of 0.6 %.

The bearing current plotted in Figure 4.6 was applied as a load. Instead of a constant stress throughout the life cycle, the following load cycles were executed until a failure occurred:

1. Reference phase and initialization for 1 h
2. Load with 10 V for 2 h
3. No load for 3 h
4. Continue with 2.

These load cycles are plotted in Figure 5.1 as the RMS of the bearing current for the overall life of one component. The actual amperage that runs through the tested bearing varies between 0.02 A at the beginning of the run ($t = 0.5$ h) to 1.6 A at $t = 1.5$ h. In addition, the amplitude during load phases varies, so that the RMS is reduced to 1.4 A at $t = 27$ h. Possible reasons for this change of electrical resistance during the life cycle of the component life cycle are manifold: burnt out lubrication, increase of bearing clearance as well as contamination of lubricant due to loss of material (cage, races). These damage cases are discussed in more detail in Section 5.1.3.

It was necessary to apply two different abort criteria in order to increase comparability between the runs. One criterion was related to the supply power of the induction machine: when the supply current in one phase exceeded an RMS value

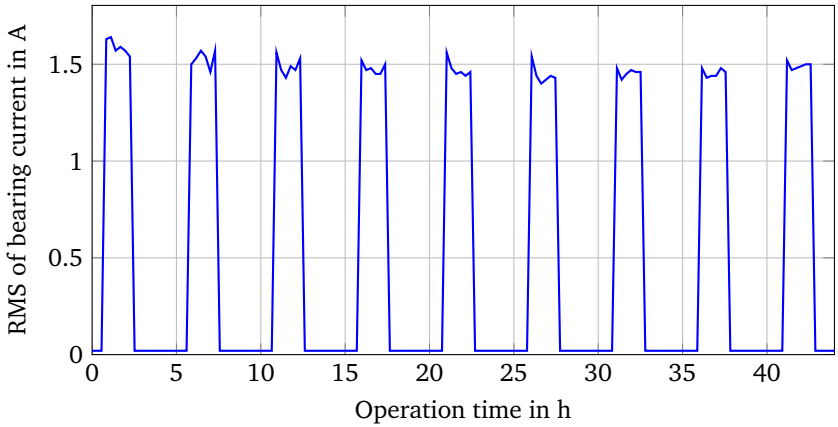


Figure 5.1.: Bearing current (RMS) during life cycle of one component

of 2.5 A, the power supply was shut down and the trial was halted. This criterion was mainly applied when the bearing cage was broken or the clearance of the bearing was so high that the rotor scratched the stator.

However, the majority of runs was stopped manually due to a strongly increased noise level in order to protect the other test rig components. Thus, this second abort criterion was based on the acceleration near the test bearing: an RMS level of 0.15 g was identified as an appropriate threshold, which on one hand was reached by the majority of trials and on the other hand correlated with the measured accelerations of the first abort criterion.

Examples of both abort criteria are plotted in Figure 5.2; whereas run 1 was interrupted by the current threshold of the first abort criterion, run 2 was stopped manually. Since the trial continued for a further twelve hours after the threshold was crossed for the first time, this time span was removed from the data and the End of Life (EoL) was assumed to be at $t_{EoL} = 60\,\mathrm{h}$. To avoid early interruptions of particular runs due to peaks as at $t = 29\,\mathrm{h}$ in run 2, a moving median filter was applied. Thus, the median of the last 20 time steps of the acceleration RMS was considered for the second abort criterion. The actual and the filtered RMS of the acceleration are plotted in Figure 5.2.

For the evaluation of the prognostic performance, several metrics were introduced in Section 2.4. Not all of them are applied, since the information contents of several metrics overlap. Thus, the prognostic horizon (PH) with the additional constraint of Equation 2.6 for prognostic convergence, the mean absolute percentage error (MAPE) for accuracy, the sample standard deviation (SSD) for prognostic

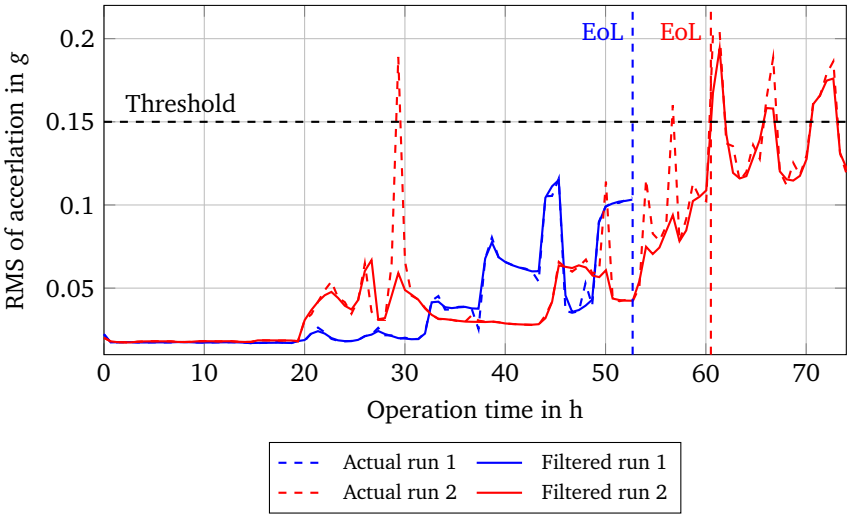


Figure 5.2.: Acceleration signal (RMS) of two different abort criteria

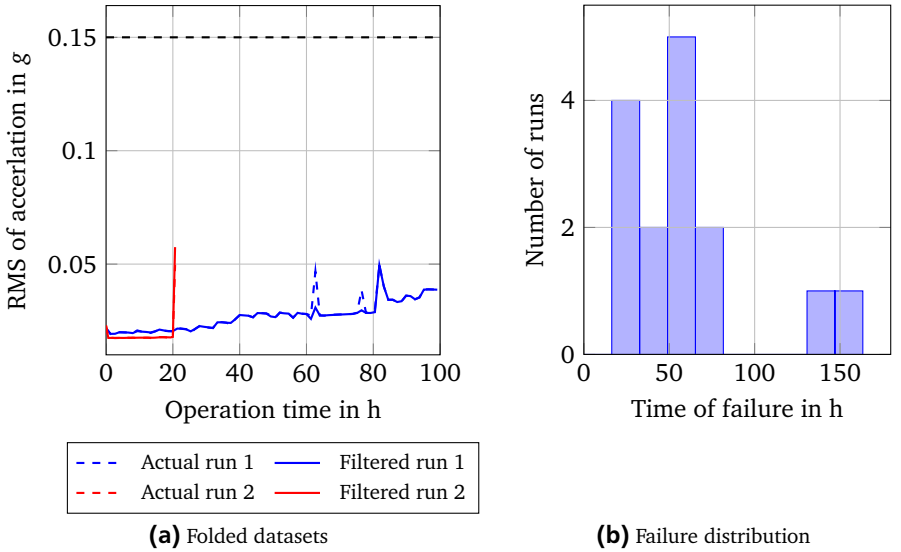
precision and the mean probability within accuracy bound (MPAB) for the assessment of prediction uncertainty are used. Additionally, the specific false detection rate (SFDR) and false negative rate (FNR) are evaluated during the validation with all datasets, since they are necessary for the cost benefit analysis in Section 5.5.2. Throughout this thesis, ten predictions of the time span between *first fault is detected* and *abort of trial* are executed. A temporally triggered prediction approach would be also possible, but this leads to an unequal number of predictions, which would influence the performance metrics of the test cases.

5.1.2 Overview of available data

A total number of 40 run to failure trials were executed with the test rig. However, due to varying bearing types (as discussed in Section 4.3) and differences in the load scheme, only 31 trials were conducted under similar conditions. A further 16 trials were abandoned, since neither of the two abort criteria introduced in the previous subsection were fulfilled, but mainly because there was no identifiable degradation before the breakdown. Both cases are illustrated by means of the RMS of the acceleration signal in Figure 5.3a. Whereas run 1 was manually stopped due to acoustic noise, run 2 was automatically switched off by the power threshold without any noticeable fault precursors. The investigation of the corresponding

filtered acceleration signal of run 1 indicated that the received RMS values did not permanently cross the threshold of 0.15 g, so that this run is not comparable to the other datasets.

Hence, 15 datasets build the database for the validation of the proposed PHM algorithm. The acceleration signals (RMS) of all datasets over their entire life cycle are illustrated in Section A.7. All datasets vary greatly with regard to degradation characteristics, initial acceleration magnitude, and total lifetime. The corresponding failure distribution is illustrated in Figure 5.3b. Whereas the majority of trials lasted 10–70 h, the life span of two other trials was about 150 h.



(a) Folded datasets **(b)** Failure distribution

Figure 5.3.: Causes of dataset reduction and resulting failure distribution of applied datasets

With the aim of finding similarities between different runs, the available datasets are clustered with reference to the monotonic behavior of the RMS values of the vibration signal. Thus, three categories are identified for this classification:

- **Strictly monotonic:** 1, 8, 11, 13, 14, 15
- **Monotonic:** 4, 10, 12
- **Not monotonic:** 2, 3, 5, 6, 7, 9

During a first validation beginning in Section 5.2, a representative subset of five datasets of all three categories (datasets 5, 7, 12, 14, and 15) is selected. Due to computational costs associated with an increasing training database (which is further investigated in Section 5.4.2) this subset is applied to show the prognostic and diagnostic performance of the proposed algorithm.

5.1.3 Achieved damage cases

Some damage cases that are the result of the run to failure tests are illustrated in Figure 5.4. The main origin of the damage cases was the bearing current and its resulting effects. In particular, the heat introduced by the electric sparks affected the components of the bearings. Thus, chatter marks on the inner and outer races of bearings were the consequence, as depicted in Figure 5.4a. This fault pattern is produced due to the combination of a steady shaft revolution and a bearing current with constant supply frequency, as mentioned in Section 4.1.2.

Additional effects of heat (which were detected in nearly all trials) were the decomposition of bearing cages and the burning out of lubrication. In some trials the degrading structure of the cage led to cracks and tears in cage material (see Figure 5.4b). In addition, dissolving cage particles and dried lubricant formed a mixture that adhered to the moving parts such as balls, and thus significantly altered the vibration and current power spectral density (PSD) spectra. Some remaining particles, still tightly adhered to the balls after the trial, are depicted in Figure 5.4c.

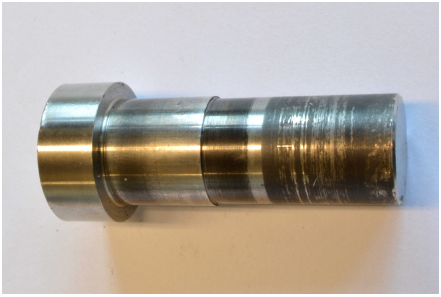
The final image of Figure 5.4d shows the shaft, which is the extension of the motor shaft (see Figure 4.3). Clear signs of wear are also visible in this picture, since many sections have changed color due to corrosion. The shaft coupling (at the right of this image), the carbon brush (right of the second shoulder), and the test bearing (right of the first shoulder) led to strong color changes and also to differences in the electric flux of the bearing current. It is assumed that due to the corrosion on the surface, on which the test bearing was mounted, the electric flux was not equally distributed across the contact surface. This led to abrupt failures of several bearings (e.g. the discarded run 2 of Figure 5.3a), so that the shaft had to be replaced.

A summary of a descriptive damage analysis of the applied run to failure trials is presented in Section A.8. By means of two tables where changes of the bearings were recorded after the trials, similar causes of failure and degradation processes can be identified. Thus, this analysis offers the opportunity to compare the defined damage states of the applied PHM algorithm with the actual detected faults.

(a) Chatter marks

(b) Torn cage

(c) Particles on balls



(d) Shaft corrosion

Figure 5.4.: Damage cases and signs of wear after run to failure trials

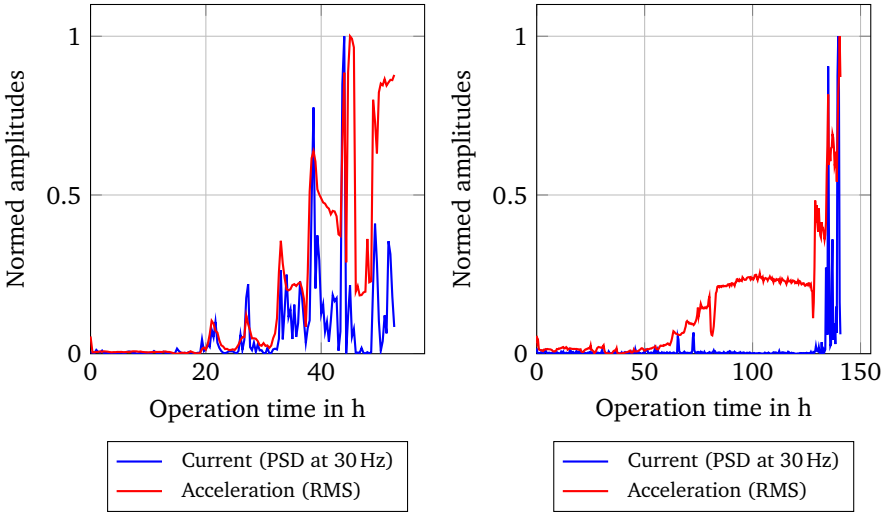
5.1.4 Resulting signals

The acceleration signal is the state-of-the-art condition monitoring signal type for bearing faults, as introduced in Section 4.1.1. Therefore, the second abort criterion and the overview of all applied datasets in Section A.7 were presented in terms of the RMS of the acceleration signal, since it is assumed to provide a more accurate representation of the current health status. However, the motor current signature analysis (MCSA) is the focus of this thesis. Thus, a comparison between both signal types of the applied data is given in this section.

As introduced in Section 2.3.2.2, one possible option is to derive features directly from signals in the time domain. For this reason, in Section A.9,the time signal of the stator current is inspected for two different points in the life cycle of an example component. Since no changes in the magnitude of the time signal can be identified, this justifies the transformation into the frequency domain. Thus, a plot similar to

Figure 3.3 is derived for each dataset. An upstream analysis of the PSD spectrum to identify an appropriate frequency range for possible features was necessary. The outcome was a range between 1 Hz and 1 kHz, which was also the specification in Section 4.1.1. Thus, all amplitudes within this range are summed up to frequency bands of 1.5 Hz in order to reduce the overall number of features.

As already stated in Section 3.2.1, the amplitudes at a frequency of 30 Hz of the resulting PSD spectrum provide the most promising health index. Hence, the amplitudes at 30 Hz of dataset 5 and 10 are examined in Figure 5.5. In addition, the RMS values of the corresponding acceleration signals are plotted. For a better comparison of both signal types, the respective amplitudes are normed.



(a) Good correspondence (dataset 5) **(b)** Poor correspondence (dataset 10)

Figure 5.5.: Correspondence of acceleration and stator current signals with regard to representation of actual health

The amplitudes of the current signal in Figure 5.5a show a good correlation to the acceleration signal. However, both signals have no monotonic behavior, since especially in the time span between 20 and 30 h the amplitudes are lowered again. The amplitudes of current signal even reach the level of the assumed nominal state. Thus, a prediction based only on this feature would be challenging, as most prediction algorithms are not able to handle a sudden "improvement" of the component. In contrast, Figure 5.5b shows a high degree of monotony, especially in the ac-

celeration signal. However, a discrepancy between the investigated signal types is also observable. Particularly in the time span between 50 and 140 h where the RMS values of the acceleration signal indicate an incipient fault, the corresponding amplitudes of the current signal remain low close to the point of failure.

Both, the lack of monotony and the discrepancy between acceleration and current signal require a multidimensional feature approach. By means of more information about the current health status of the component covered by amplitudes of other frequencies, benefits with regard to diagnostic and prognostic accuracy are assumed. This hypothesis is evaluated in Section 5.3.5.

Finally, a comparison of one resulting (continuous) feature for four applied training datasets is presented in Figure 5.6. In accordance with Figure 3.2, this feature is the outcome of feature generation and subsequent feature reduction with respect to the mutual information criterion. For better visualization, the continuous feature of $\mathbf{\Phi}_c$ instead of the discretized feature $\mathbf{\Phi}$ is plotted. As expected, all trained features begin and end on a similar amplitude level, and are therefore appropriate for distinguishing different damage states. After discretizing all features, the resulting feature matrix $\mathbf{\Phi}$ is transferred to the diagnosis step.

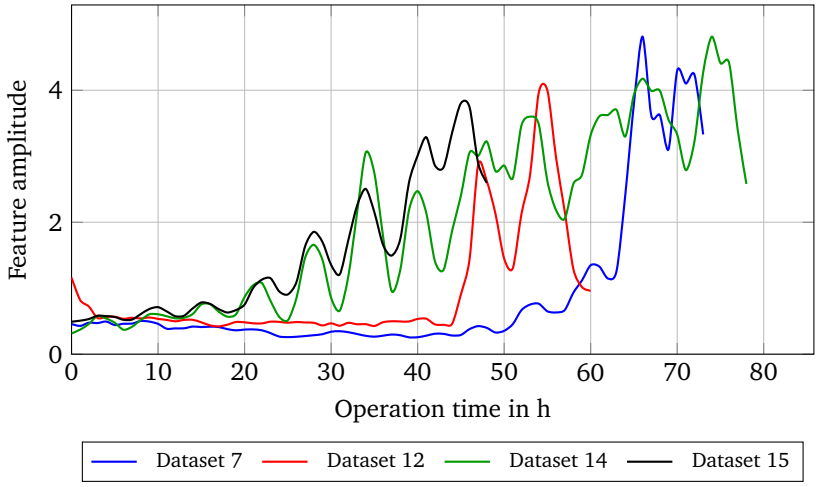


Figure 5.6.: One selected feature of training during the test with dataset 5

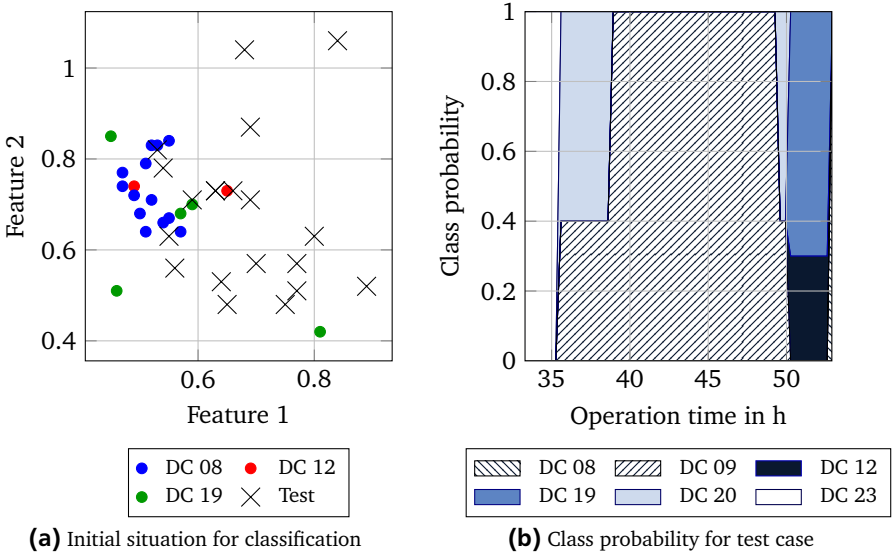
5.2 Diagnosis results with selected data subset

The proposed algorithm is validated with the database of the previous section. As mentioned and motivated in Section 5.1.2, the first results are achieved by a leave one out cross validation (LOOCV) with datasets 5, 7, 12, 14, and 15. Thus, four runs are used for training and the remaining dataset is applied as a test case. The feature generation and training process of K Nearest Neighbor (KNN) and Hidden semi-Markov Models (HsMM) is skipped in the context of this validation, since it was introduced in detail in Section 3.2 and during the verification in Section 3.6 with the applied data. Thus, the validation begins with the fault diagnosis step. Since the two proposed PHM approaches, probability-based and load-based Hidden semi-Markov Models (HsMMs), are based on the same results in fault diagnosis, this section is also separated from the subsequent sections, which focus on the prognostic step.

The fault diagnosis begins with the classification of the test dataset by the trained KNN, as illustrated in Figure 3.2. Here, the three-dimensional feature subset $\boldsymbol{\Phi}$ is reduced to the unidimensional damage class vector $\boldsymbol{C}$. In addition, the class probability $\boldsymbol{S}_C$ is calculated in accordance with Equation 3.3. The classification step is presented in Figure 5.7. Whereas the distribution of training and test data points is illustrated in Figure 5.7a, one example of the resultant damage class probability $\boldsymbol{S}_C$ is presented in Figure 5.7b. The circles in Figure 5.7a represent the already classified training points and the crosses correspond to the new test points, which must be assigned. In contrast to Figure 3.16, the test points and the training points are not equal in the context of validation during the verification. With respect to the distance to adjacent training points, the test points are classified, so that in border areas between the trained classes the class assignment is not unique. Thus, the outcome is a class probability $\boldsymbol{S}_C$ for every test data point depicted in Figure 5.7b. The results can be interpreted as follows: in $t = 36\,\mathrm{h}$, 40 % of $K = 5$ surrounding training points belong to damage class 8 and the rest is assigned to damage class 20.

The matrix $\boldsymbol{S}_C$ is then transferred to the state estimation algorithm, presented in detail in Section A.4. For a better comparison of estimated states and trained state sequences, a directed graph of the HsMM net, comparable to Figure 2.15, is illustrated in Figure 5.8. This provides a better overview of possible damage state transitions within the HsMM net, which is trained with datasets 5, 7, 12, and 14. The related damage classes and the damage states which are the basis of this directed graph are presented in Section A.10.1. Several discoveries can be derived from this plot and Figure 5.8:

- All trained models start in S_1 (new/nominal state)



(a) Initial situation for classification

(b) Class probability for test case

Figure 5.7.: Classification of a new test dataset by means of the KNN algorithm (DC=Damage Class)

- Several damage states are entered by different models (e.g. S_1, S_4, S_8 etc.)
- Although some internal cycles between two damage states exist (e.g. $S_1 \longleftrightarrow S_2$, $S_{11} \longleftrightarrow S_{13}$), the general characteristic of state transitions is acyclic, i.e. there are no transitions from damage states close to end of life cycle to beginning of life cycle (e.g. $S_{12} \rightarrow S_1$)
- The models of datasets 5 and 12 show a similar fault process, since both models include the states S_1, S_4, and S_9

As mentioned in Section 5.1.3, based on the derived damage states and the descriptive damage analysis of Section A.8, a comparison between datasets is possible. Datasets 5 and 12 show similarities in the context of the descriptive damage analysis with regard to roughening of bearing components, burnt out lubrication, or especially the cage condition at the end of the trial. However, the divergence in the selected abort criterion or the appearance of chatter marks in dataset 5 indicate that these similarities can be coincidental.

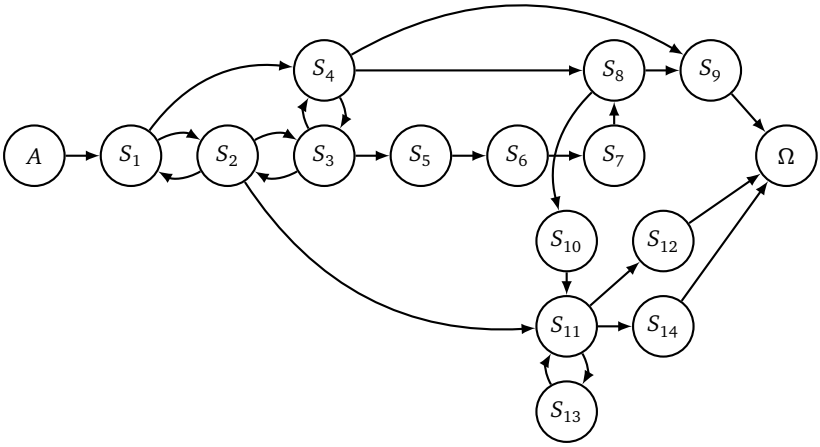
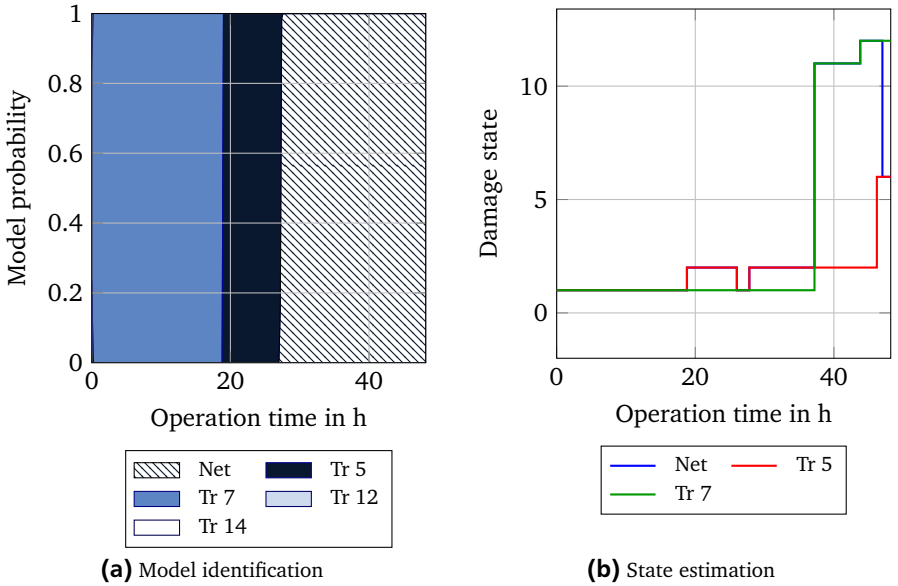


Figure 5.8.: Real net structure with trained damage state transitions of training datasets 5, 7, 12, 14

The final step of fault diagnosis consists of model identification and damage state estimation, which are both illustrated in Figure 5.9. In combination with Figure 5.8 and Figure A.7 of Section A.10.1, the results can be explained as follows: at the beginning of the life cycle of the tested bearing, the model trained by dataset 7 is selected as most suitable in Figure 5.9a. As described in Section 3.6.3, this stems mainly from the trained emission matrix $\boldsymbol{B}$. At $t \approx 20\,\mathrm{h}$, the model, derived from dataset 5, has the highest model probability, since the assumed state transition in Figure 5.9b for model 5 from S_1 to S_2 is captured in this model. A final change in model identification takes place at $t \approx 26\,\mathrm{h}$, when another state transition from S_2 to S_1 is assumed by model 5 and the net; the net is subsequently selected as the most probable model. This stems from the duration time within damage state S_2 derived by the state estimation: since the estimated duration time $d(S_2) \approx 26{-}20\,\mathrm{h}$ within damage state S_2 does not fit with the trained duration times of model 5 ($d(S_2) < 4\,\mathrm{h}$), the net is selected as the most probable. Thus, the net combines both the possible state transitions and a suitable duration time within S_2.

A final remark is related to the estimated damage states at the end of the life cycle in Figure 5.9b: until $t \approx 45\,\mathrm{h}$, the selected models indicate state transitions, which are covered by the directed graph of Figure 5.8. Thereafter, the assumed state transitions differ from the trained state sequences (net: $S_{12} \rightarrow S_6$, model 5: $S_2 \rightarrow S_6$). This divergence between actual damage state sequence and the sequence propagated by the models reveals that not all possible fault propagations

are covered by the models. Hence, this inevitably leads to inaccuracies in fault prediction.



(a) Model identification **(b)** State estimation

Figure 5.9.: Model identification and state estimation during LOOCV with tested dataset 15 (Tr=Training dataset)

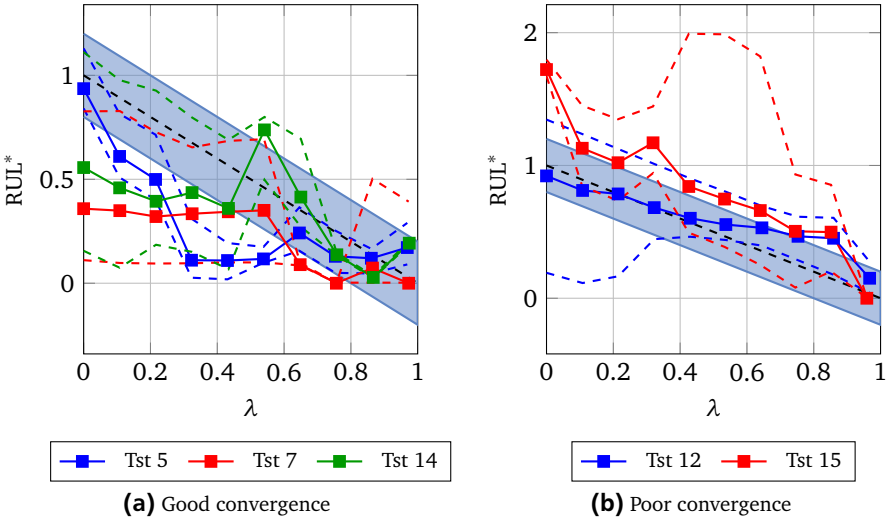
5.3 Prediction results with selected data subset

The prognostic performance of the two proposed algorithms as well as the comparative algorithm is validated in this section. As motivated in Section 5.1.2, the algorithms are tested with a data subset consisting of datasets 5, 7, 12, 14, and 15. The selected validation method is a leave one out cross validation (LOOCV), where one dataset is used to test the HsMMs that are trained with the remaining datasets.

First, the two proposed algorithms, which differ in the state duration estimation introduced in Section 3.5.1, are validated; the probability-based method is followed by the load-based duration estimation. The comparative HsMM algorithm of Section 3.7 is then tested. During the final three subsections, the impact on the prognostic performance of different parameter settings and a change from stator current to vibration signal is investigated.

5.3.1 Probability-based approach

Based on the results of the fault diagnosis with regard to state estimation and model identification, the fault prognosis is initiated. As introduced in Section 3.5.1, this probability-based method is characterized by a damage state duration estimation based on a probability distribution. The prognostic results of the LOOCV produced with this approach are depicted in Figure 5.10. Therefore, the five prediction sets are split into trials with a higher prognostic convergence expressed by the prognostic horizon (PH) in Figure 5.10a and a poor convergence in Figure 5.10b.



(a) Good convergence **(b)** Poor convergence

Figure 5.10.: Prediction results of probability-based HsMM approach in context of LOOCV with datasets 5, 7, 12, 14, 15 (Tst=Test dataset)

A noticeable conformity among the predictions of Figure 5.10a is that in an area between $\lambda = 0-0.4$, all predictions tend to conservative forecasts. This is, because trial 7 and 14 are the longest runs (72 and 78 h) in the selected subset. Thus, the HsMMs trained with the shorter remaining datasets drift to false positive predictions. Despite the good PH, especially for test 5 and 14, the prediction sets have problems related to prognostic accuracy and uncertainty. In Table 5.1, the corresponding performance metrics are summarized. The prognostic accuracy, expressed with mean absolute percentage error (MAPE), of test 14 is 108 %, and in addition

only 18 % on average of the predicted RUL distribution of test 7 lies within the defined accuracy bound, covered by mean probability in accuracy bound (MPAB).

In contrast, the MAPE values of the prediction sets in Figure 5.10b are low compared to test 14. In particular, the final prediction of both tests fits the actual RUL almost perfectly. In addition, the prediction precision, covered by the sample standard deviation (SSD), is small for these trials due to nearly constant prediction errors. The low MPAB value of test 15 is a result of the wide RUL spread between $\lambda = 0.3 - 0.7$.

Table 5.1.: Performance metrics of probability-based HsMM approach

Test	PH [λ]	MAPE [%]	MPAB [%]	SSD
5	0.35	86	50	138
7	0.14	62	18	152
12	0.03	76	49	78
14	0.35	108	53	198
15	0.04	82	23	95

The basis for these predictions during this first approach is a duration probability density distribution (PDF), as in Figure 5.11 for damage state S_1. As formulated in Section 3.5.1, these distributions determine the duration time within each damage state until the failure state Ω is reached. These are derived during the training process, when the particular damage state is entered. In Figure 5.11, state S_1 was selected eight times with a wide spread of duration times from $d(S_1) = 4 - 33\,\text{h}$. This results in a large prediction uncertainty at low λ of test 7 in Figure 5.10a. This is assumed to be reduced with the load-based approach, which is the focus of the following subsection.

5.3.2 Load-based approach

Instead of a probability distribution of the duration time within each damage state as in the previous subsection, the sojourn time of the load-based approach is estimated by the model, depicted as an example in Figure 3.12. Analogously to the probability-based HsMM approach of the previous section, this method is validated by means of a LOOCV with datasets 5, 7, 12, 14, and 15. Figure 5.12 is again separated into predictions with a higher convergence (Figure 5.12a) and forecasts where the predicted and actual RUL do not converge (Figure 5.12b).

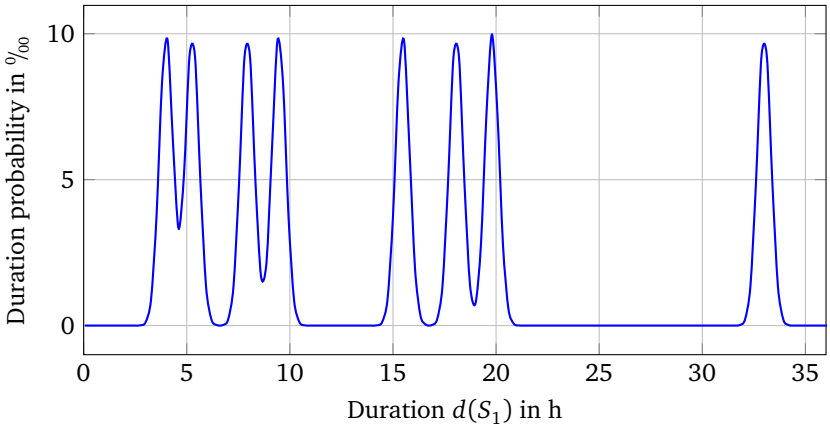


Figure 5.11.: Example duration probability within damage state S_1 (training datasets 5, 12, 14, and 15)

All three trials shown in Figure 5.12a predict the particular RUL of the tested component at the beginning of the component's life cycle with high accuracy, although the actual RUL of all participating runs vary greatly. Whereas in test case 5 the predictions tend to be conservative between $\lambda = 0.10 - 0.55$, tests 12 and 14 match the required PH accuracy over the entire life cycle, so that even in the first three predictions of test 14 the majority of probability mass lies within the accuracy bound. The constant prediction error of test case 12 stems from the model identification and state estimation: over the entire prediction phase, only one model and also one damage state are identified as the most suitable, so that based on this, there are no changes in the predictions.

The low prediction errors for test cases 12 and 14 also manifest in small values in the accuracy-specific metric MAPE as well as the precision assessing metric SSD. All metrics are summarized in Table 5.2. The trends in prediction accuracy are also reflected in the values of MPAB; since the majority of estimated RUL probability distribution lies within the accuracy bound, the values for MPAB are high. In contrast, the metrics for test cases 7 and 15 in Figure 5.12b are significantly worse. The PH in particular is reduced due to the prediction errors near the end of the component's life cycle. Additionally, the spread of the predicted RUL probability distribution is considerably larger in comparison to test cases 5, 12, and 14. This leads to reduced MPAB values in both cases.

As introduced in Section 3.5.1, the estimation of the duration time within a damage state in this approach uses the cumulative load u_Σ as additional input.

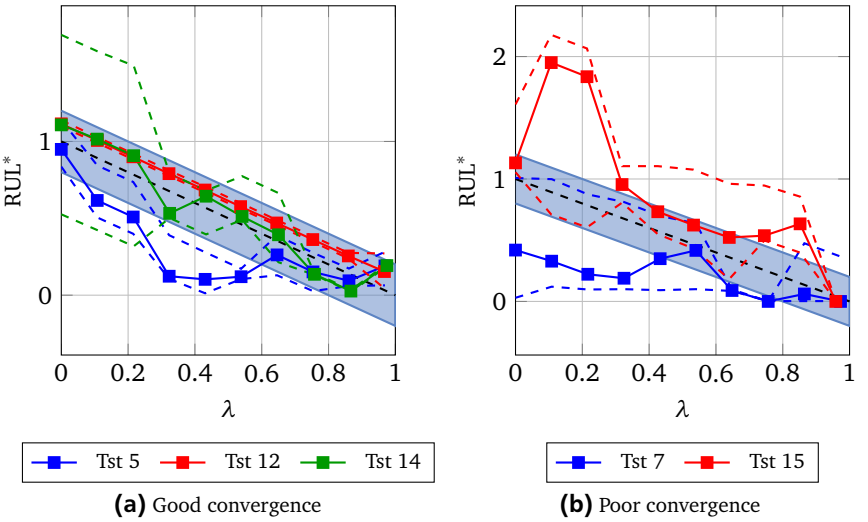
Table 5.2.: Performance metrics of load-based HsMM approach

Test	PH [λ]	MAPE [%]	MPAB [%]	SSD
5	0.35	92	49	139
7	0.14	64	23	166
12	1.00	63	96	1
14	1.00	88	76	88
15	0.04	95	26	186

By means of linear interpolation and extrapolation between derived training data points, the duration time is calculated for new test points when the particular damage state is entered. Thus, in Figure 5.13 three examples of the cumulative bearing current u_Σ (Figure 5.13a) as well as the resulting duration model (Figure 5.13b) are illustrated. The cumulative load is derived by integrating the load phases of Figure 5.1. In the case of prediction, when the future cumulative load is not available, several approaches are possible. If the expected load profile is not assessable, u_Σ must be predicted. For this first approach, it is assumed that the load profile is available, and the actual load is therefore applied. Although all bearings were loaded with the same voltage, differences in the resulting u_Σ arise due to varying electrical resistance of the lubricant in the bearings for example, as described in Section 4.1.3. In Figure 5.13a for training dataset 12, the effect of the lubricant is visible, since a significant increase in u_Σ is registered after 15 h, when the majority of lubricant was burnt out.

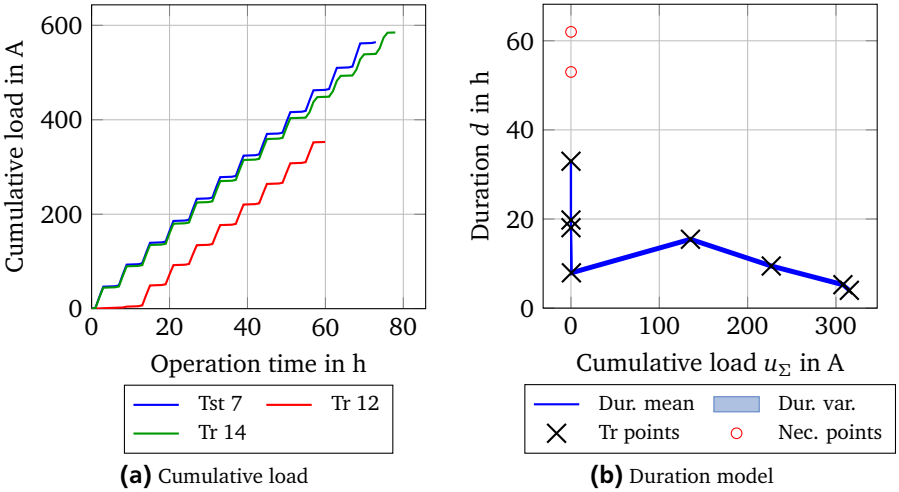
In Figure 5.13b, the duration model based on u_Σ as input is plotted. As described in Section 3.5.1, the training points are generated, when the particular damage state is entered during the training phase. Thus, every time the state is entered, the current cumulative load and the corresponding duration time in this state is applied as training inputs and outputs, respectively. The resulting model shows an approximate hyperbolic behavior with smaller values at larger cumulative loads.

This model can also be connected to the predictions of test case 7 in Figure 5.12b. Especially at the beginning, the predictions are conservative. This can be explained by the duration model: when damage state S_1 is entered at a low value of u_Σ, the maximum duration time of the trained model does not exceed $d = 38\,\text{h}$. Based on the state estimation of test case 7 (given in Section A.10.2), a duration time of $d = 52\,\text{h}$ in damage state S_1 would result in perfect predictions for this state. However, since the maximum duration of the model is $d = 38\,\text{h}$, this inevitably leads to conservative predictions until $\lambda = 0.45$.



(a) Good convergence **(b)** Poor convergence

Figure 5.12.: Prediction results of load-based HsMM approach in context of LOOCV with datasets 5, 7, 12, 14, 15



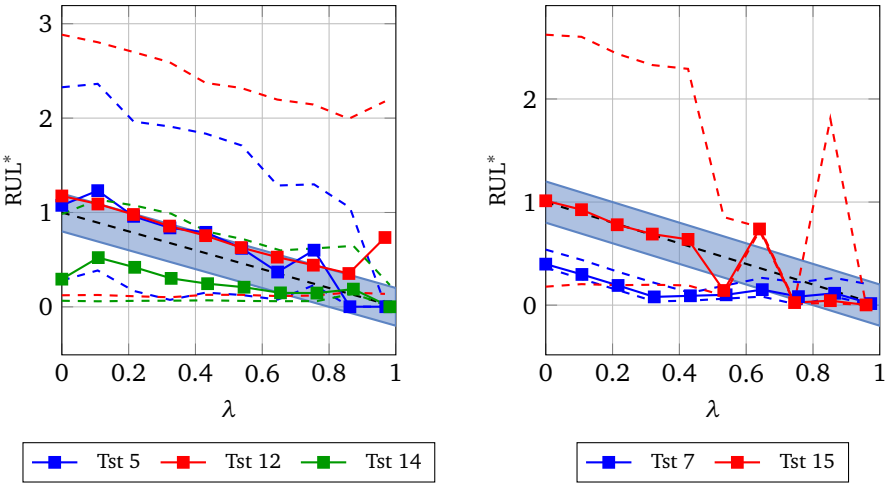
(a) Cumulative load **(b)** Duration model

Figure 5.13.: Determination of duration in damage state S_1 by means of cumulative load u_Σ in context of LOOCV (training datasets 5, 12, 14, and 15)

5.3.3 Comparative approach

The results of the LOOCV in the case of the comparative approach, (introduced as a benchmark method in Section 3.7) are plotted in Figure 5.14. Several differences to the results of the previous approaches can be identified in the performance metrics of Table 5.3. The prediction accuracy is higher in the majority of trials, which leads to lower MAPE values. This can be also derived by examining Figure 5.14: the prediction errors are small and the predictions show a good convergence at the end of the life cycle. The test case 15 in Figure 5.14b in particular is predicted very accurately at the beginning and at the end, so that a MAPE value of 45 % is achieved.

However, the prediction uncertainty is higher in the majority of trials, which reduces the MPAB values. This stems from the internal cycles within the trained HsMM, which produce the approach-dependent state transitions between the assumed states *nominal* S_1, *incipient fault* S_2, and *failure* S_3. Test case 12 in particular shows a wide probability spread over the entire life cycle of $\mathrm{RUL}^*(\lambda = 0.01) = [0.1,2.9]$. In addition, although the convergence of the fault predictions is good, the achieved PH values are in general lower in comparison to the results in Table 5.2.



(a) Test cases analogous to Figure 5.10a

(b) Test cases analogous to Figure 5.10b

Figure 5.14.: Prediction results of comparative HsMM approach in context of LOOCV with datasets 5, 7, 12, 14, 15

Table 5.3.: Performance metrics of comparative HsMM approach

Test	PH [λ]	MAPE [%]	MPAB [%]	SSD
5	0.14	51	16	103
7	0.24	64	28	181
12	0	258	30	100
14	0.35	65	30	241
15	0.15	45	16	91

5.3.4 Net probability

In the following subsections, several parameters of the proposed HsMM approach are varied in order to investigate their impact on the prognostic performance. Since the load-based duration estimation approach achieved the best results with the selected subset of datasets 5, 7, 14, and 15 with regards to prediction accuracy, convergence, and precision, this will be the focus of this investigation.

In Section 3.4.2, the possibility of overruling the likelihood-based model identification by manually selecting the net probability $P(m_{net})$ was introduced. In this way, it is guaranteed that a constant ratio of prediction samples stems from the HsMM net. Analogous to the verification in Section 3.6.5, the analyzed net probabilities are $P(m_{net}) = [0\,\%, 30\,\%, 70\,\%, 100\,\%]$.

In Figure 5.15, the evolution of MAPE and PH for the selected net probabilities is presented. Besides the actual sampled values, the arithmetic mean is also plotted in order to highlight the overall trend of the particular performance metric. Both metrics indicate a similar development: predictions with a low net probability result in high values for MAPE and low PHs, whereas with increasing $P(m_{net})$ the prognostic accuracy and convergence also rise. The spread of MAPE values is also strongly reduced for $P(m_{net}) > 30\,\%$.

All investigated averaged performance metrics for the analyzed net probabilities as well as the likelihood-based net probability (named as *variable*) are summarized in Table 5.4. In comparison to MAPE and PH, the MPAB values remain approximately constant for all net probabilities (except for the likelihood-based), whereas the SSD values follow the trend of higher prognostic precision with increasing $P(m_{net})$.

One finding that could be derived from this overall trend is that likelihood-based model identification provides an optimal solution, since the likelihood-based ap-

proach is outperformed by $P(m_{net}) = 100\,\%$ only in SSD. It is noticeable by examining the resulting likelihood-based model probabilities, which are plotted for further interest in Section A.10.3, that the net probability is 1 in four of five cases, especially at the end of the runs. This could be a reason for the increase of PH values in correspondence to larger $P(m_{net})$.

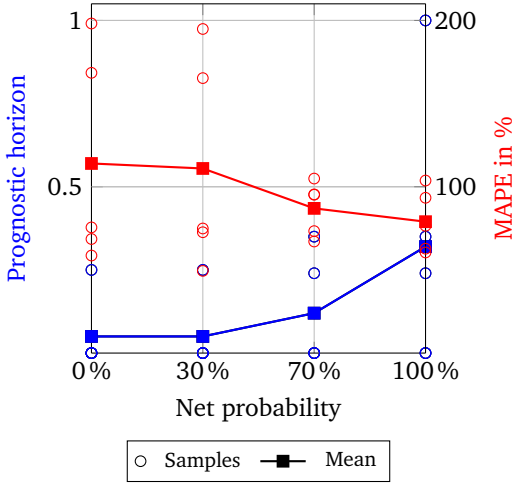


Figure 5.15.: Evolution of MAPE and PH with increasing ratio of net probability $P(m_{net})$

Table 5.4.: Averaged performance metrics with constant and variable (likelihood-based) net probability $P(m_{net})$ (load-based HsMM method)

Net probability	$\overline{\text{PH}}$ [λ]	$\overline{\text{MAPE}}$ [%]	$\overline{\text{MPAB}}$ [%]	$\overline{\text{SSD}}$
0 %	0.05	113	40	168
30 %	0.05	111	42	151
70 %	0.12	87	46	124
100 %	0.32	79	45	96
Variable	0.51	80	54	116

5.3.5 Number of features

The choice of feature dimension has an even greater impact on prognostic performance compared to the selected net probability. While the default for a database consisting of 4 training datasets is set to 3 features, the performance with 1 up to 5 features is analyzed in this section. In Section A.10.3, the corresponding features are plotted.

The trend of PH and MPAB for an increasing number of features is plotted in Figure 5.16. The PH values indicate that a number of features differing from the default value 3 result in significantly worse convergence. A symmetry with respect to the default value can be identified, since 2 and 4 as well as 1 and 5 features create comparable results. A similar characteristic is detected in the case of MPAB. Here, a feature set consisting of 3 features also shows the best performance, whereas the symmetry of the PH values is not given in this case. However, although the arithmetic mean generated by the default number of features results in the best prognostic performance, the spread of the PH and MPAB samples is also increased, so that the PH values for example are distributed from PH= 0.03 to PH= 1.00.

In Table 5.5, all (averaged) achieved performance metrics are summarized. Similar results to the PH and MPAB can also be detected for MAPE and SSD. While a set of features less than 3 might contain insufficient information about the degradation process, a feature dimension greater than 3 could lead to the curse of dimensionality, which was introduced in Section 2.3.2.3. Thus, the correct selection of the feature dimension is a crucial part of the proposed algorithm and requires an intense upstream analysis.

Table 5.5.: Averaged performance metrics with increasing number of features (default= 3)

No. of features	$\overline{\text{PH}}$ [λ]	$\overline{\text{MAPE}}$ [%]	$\overline{\text{MPAB}}$ [%]	$\overline{\text{SSD}}$
1	0.01	171	35	220
2	0.07	158	20	189
3	0.51	80	54	116
4	0.09	177	39	203
5	0.01	177	28	226

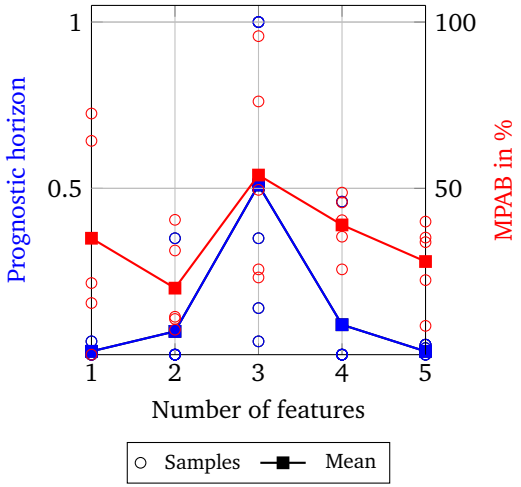


Figure 5.16.: Evolution of MPAB and PH with increasing number of features (default= 3)

5.3.6 Stochastic robustness

Since many steps during model training and fault prediction contain stochastic functions, examination of prognostic robustness against stochastic influences is the focus of this subsection. During model training, the clustering with K-means in particular can vary, since the initial distribution of clusters is arbitrarily selected, as discussed in Section 3.3.1. This further provides the required input and output of K nearest neighbor, and can therefore change the inputs to HsMM training for example, which influences all subsequent processes.

In order to generate reproducible results despite stochastic effects, MATLAB provides so-called *seeds* to initialize the internal random number generator. Thus, by selecting the same seed, the generated random numbers become predictable. In Figure 5.17, the prognostic performance for three different seeds for test dataset 15 is illustrated. Therefore, seed 1 corresponds to the testing of Figure 5.12b. As expected, there are stochastic influences on the prognostic performance, since the prediction of seed 2 differs from seed 1 for instance. However, the influences are minor, since for example seed 3 covers seed 1 over the entire prediction phase.

Due to stochastic effects, the performance metrics also vary. As depicted in Table 5.6, they fluctuate in a range of up to 12 % in the case of prognostic precision

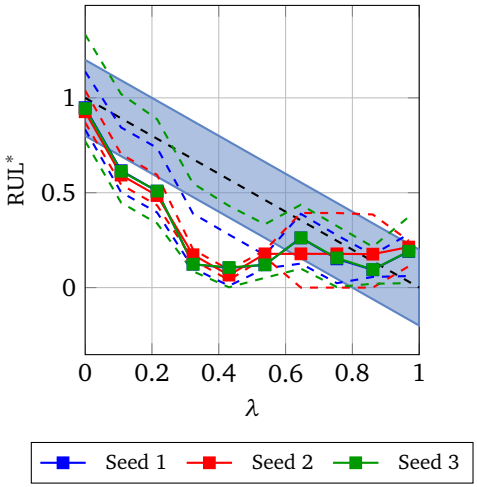


Figure 5.17.: Changes in prognostic performance due to stochastic effects with test dataset 15

SSD. Other metrics such as PH vary by about 6 %, so they are more robust against stochastic influences.

Table 5.6.: Averaged performance metrics with different seeds (load-based HsMM method)

Seed	$\overline{\text{PH}}$ [λ]	$\overline{\text{MAPE}}$ [%]	$\overline{\text{MPAB}}$ [%]	$\overline{\text{SSD}}$
1	0.51	80	54	116
2	0.51	86	53	121
3	0.48	79	57	133

5.3.7 Analysis of vibration signal as input data

Although the focus of this thesis is on the MCSA, the proposed algorithm is intended to process any signal type of rotating components. Thus, this section investigates the prognostic performance of the load-based HsMM approach using the vibration signal as input data.

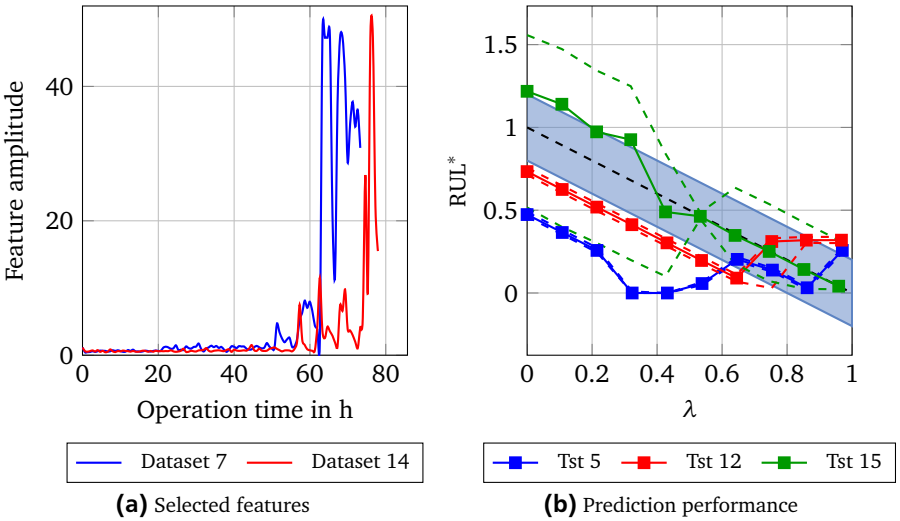
In addition to the prognostic performance (Figure 5.18b), one selected feature for two datasets during the training phase is also presented in Figure 5.18a. Many similarities to the feature in Figure 5.6 with stator current as input can be derived: in both datasets the feature begins and ends at a comparable amplitude, and is also not strictly monotonic. However, an increasing trend is clearly visible, especially in the last third of the life cycle of both components. Thus, from the point of view of the features, both signal types are analogous.

The prognostic performance with vibration signal as input data (plotted in Figure 5.18b) also indicates similarities with the results of the stator current in Figure 5.12: whereas the prediction of test case 5 shows an analogous behavior with regard to convergence and accuracy, the forecasts of test case 15 with vibration data even outperform the results with stator current with PH$= 0.58\lambda$. The trend of comparable results is also reflected in the averaged performance metrics (summarized in Table 5.7) for both vibration and stator signal. The depicted MAPE values are on a similar level, and in addition the MPAB is only slightly worse in the case of vibration data, so that the prognostic accuracy of both signal types is analogous. In contrast, the SSD and PH values indicate a significant performance decrease. The PH values in particular are reduced, since the last prediction probabilities with vibration signals lie outside the assumed accuracy bound.

Table 5.7.: Comparison of averaged performance metrics for signal type vibration and stator current (load-based HsMM method)

Signal	$\overline{\textbf{PH}}$ **[λ]**	$\overline{\textbf{MAPE}}$ **[%]**	$\overline{\textbf{MPAB}}$ **[%]**	$\overline{\textbf{SSD}}$
Vibration	0.19	84	41	146
Current	0.51	80	54	116

The results of this subsection demonstrate that the application of stator current leads to similar results in comparison to the state-of-the-art vibration signal with regard to feature generation and prognostic performance. For some test cases, the stator current-based forecasts even outperform the results with vibration data. Thus, due to possible cost reductions, a recommendation can be made for an MCSA based PHM system.



(a) Selected features **(b)** Prediction performance

Figure 5.18.: Feature selection and prognostic performance of load-based HsMM approach in context of LOOCV (with datasets 5, 7, 12, 14, and 15) with vibration signal as input

5.4 Prediction results with all datasets

Instead of the selected subset in the previous section, a validation with all 15 datasets is executed in this section. The database is arbitrarily divided into training and test data. For a better comparison, all examined HsMM approaches are trained and tested with the same randomly selected datasets. The number of training datasets N_{td} is 4, 6, 8, and 10; for every N_{td}, 5 different trainings are performed. The trained HsMMs are then tested on 5 also randomly selected test datasets, so that in total 100 prediction sets are generated for each HsMM approach. The focus of this validation is on the three approaches: probability-based, load-based, comparative. In addition, the results with differing net probability $P(m_{net})$ in the case of the load-based method is examined.

The section begins with an overall comparison of the prognostic performance that is achieved by the considered HsMM approaches. The following subsection focuses on the influences and especially the robustness of the prognostic performance with varying amounts of training data.

5.4.1 Overall comparison

The validation results scored by the selected approaches are summarized in Table 5.8. Again, the arithmetic mean of each performance metric is presented. Besides the three HsMM methods of Section 5.3.1 to Section 5.3.3, the load-based HsMM approach with $P(m_{net}) = 0$ and $P(m_{net}) = 1$ are also examined, as motivated in Section 5.3.4.

In comparison to the results with a selected subset of datasets, nearly all performance metrics are considerably worse. The values of PH in particular are strongly reduced to only PH$= 0.10\lambda$ for all proposed HsMM approaches and PH$= 0.05\lambda$ in the case of the comparative HsMM concept. Thus, all approaches have difficulty identifying the true RUL in the majority of test cases. Selected predictions for all three HsMM approaches are presented in Section A.10.4.

There are specific differences between the proposed methods and the comparative HsMM approach. The prognostic convergence and accuracy of the proposed methods in particular are significantly better. The MPAB and SSD values of all approaches are comparable. A clear difference between the results of the probability-based and load-based HsMM method cannot be recognized, although the metrics of the load-based approach indicate slightly improved results for all categories except PH. The influences of the selected model probability $P(m_{net})$ are also noticeable, since predictions which only stem from the net ($P(m_{net}) = 1$) result in more accurate and precise forecasts.

Table 5.8.: Averaged performance metrics for the three examined HsMM approaches and a fixed net probability (also load-based HsMM approach)

Approach	$\overline{\text{PH}}$ [λ]	$\overline{\text{MAPE}}$ [%]	$\overline{\text{MPAB}}$ [%]	$\overline{\text{SSD}}$
Prob. HsMM	0.10	173	24	304
Load. HsMM	0.10	166	26	298
Comp. HsMM	0.05	271	26	283
$P(m_{net}) = 0$	0.10	172	25	285
$P(m_{net}) = 1$	0.10	162	25	278

For a better visualization, the metrics of Table 5.8 are plotted in a radar chart in Figure 5.19. For this purpose, the metrics were normalized by min-max-scaling (see Section 2.3.2.1), so that the results are more distinct. In addition, metrics which are optimal in the case of low values such as MAPE or SSD are transformed ($\text{MAPE}^* = 1 - \text{MAPE}^{*-}$, where MAPE^{*-} represents the normalized value

before transformation), so that the best results are achieved with a high percentage. The chart shows that none of the approaches outperforms the others. However, the load-based HsMM approach with likelihood-based model probability and with a net probability of $P(m_{net}) = 1$ achieves good results for the majority of examined performance metrics. This can be derived by inspecting the area covered by each approach within the radar chart. According to this illustration, the probability-based approach shows the poorest performance. However, the performance metrics of Table 5.8 show only slight differences to the load-based approach for example.

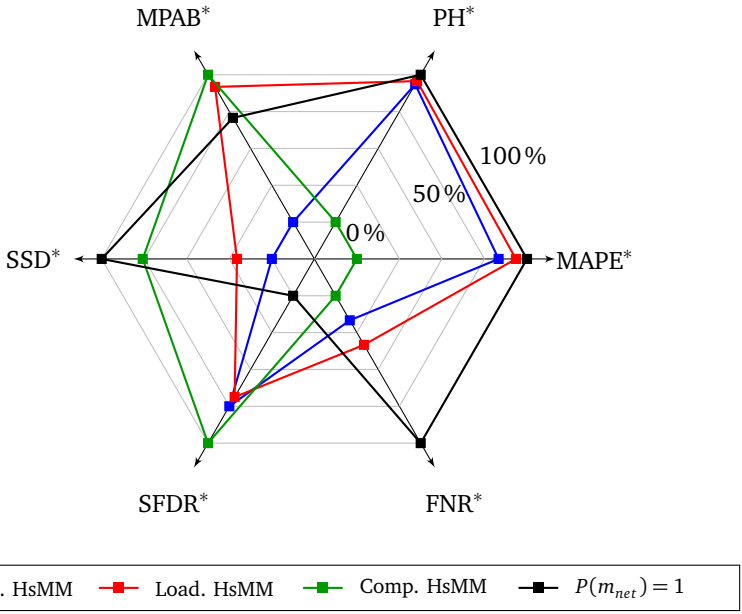


Figure 5.19.: Comparison of four HsMM approaches with normalized performance metrics

In addition to the performance metrics in Table 5.8, the false negative rate (FNR) and specific false discovery rate (SFDR) of Equation 2.9 and Equation 2.10 are also plotted. Here, a time for maintenance action $\Delta t_{ma} = 10$ steps is selected. These values will be important in Section 5.5.2, where the cost benefits in the case of an actual implementation of the proposed approaches are evaluated. The values for SFDR and FNR are illustrated in Table 5.9. Besides the two rates, the achieved true positives (TP), false positives (FP), and false negatives (FN) of the

100 prediction sets are also presented. All resulting FNR values are comparable to each other, and only the SFDR of the comparative HsMM is slightly lower compared to the proposed algorithms. However, due to the prediction inaccuracies during this second validation, the achieved error rates (especially in the case of FNR) are also high.

Table 5.9.: Achieved SFDR and FNR for the three investigated HsMM approaches

Metric	TP	FP	FN	**SFDR** in %	**FNR** in %
Prob. HsMM	3	21	76	21	96
Load. HsMM	4	22	74	22	95
Comp. HsMM	2	17	81	17	98

A discussion of all achieved prognostic performances is presented in Section 5.5.1. Besides the differences in the examined HsMM approaches, reasons for the comparatively poor prognostic performances of all approaches are also provided.

5.4.2 Amount of training data

The validation of the HsMM approaches by means of all datasets allows the assessment of the influences of a varying amount of training data N_{td}. As introduced in the preamble of this section, the evaluated N_{td} are 4, 6, 8, and 10, which is mainly motivated by the number of available datasets and the process time to generate one prediction. Therefore, the averaged time (per test run) allocated to model training, state estimation and model identification, and actual prognosis with respect to N_{td} are illustrated in Table 5.10. Here, only the load-based HsMM and the comparative HsMM approaches are examined, since the probability-based HsMM method is identical to the load-based algorithm in terms of duration for training and state estimation. The process time for one prediction is also similar in both proposed methods. The comparison of load-based and comparative approach indicates only slight differences in prediction process time. The training duration for small values of N_{td} is also similar. For increasing N_{td}, the time for training and especially for state estimation rises exponentially in the case of the proposed approaches. This is caused by an increasing model complexity, since more damage states are necessary in order to cover all available feature combinations. Therefore, the dimensions of the majority of HsMM parameters (transition matrix $\boldsymbol{A}$, emission matrix $\boldsymbol{B}$, duration

matrix $\boldsymbol{D}$) rise, which leads to longer computing times. Thus, N_{td} was restricted to a maximum of 10.

Table 5.10.: Averaged process time [min] with respect to the varying number of training data N_{td}

	Training				**State Estim.**				**Prognosis**			
No. Tr. Data →	4	6	8	10	4	6	8	10	4	6	8	10
Load. HsMM	4	9	24	40	7	16	43	95	8	7	13	8
Comp. HsMM	8	13	10	12	5	8	12	21	10	7	8	6

Prognostic performance with varying N_{td} is illustrated in Figure 5.20. The evolving prognostic accuracy (Figure 5.20a) as well as the prognostic precision in Figure 5.20b are presented in the form of arithmetic mean and corresponding 5th and 95th percentile (in dashed lines). The results indicate a large spread for all applied HsMM approaches in both, accuracy and precision. The mean of MAPE values for the probability-based and load-based approach is relatively constant for the four examined N_{td}, whereas the values in the case of the comparative algorithm converge to a similar accuracy level at $N_{td} = 10$. Thus, an increase of training data for the proposed algorithms is not necessarily beneficial. In contrast to the accuracy, the averaged precision metric of Figure 5.20b indicates a rather robust behavior in the case of the comparative algorithm and a slight trend to decreased precision with rising N_{td} for the other two approaches.

By means of these values, a quantitative assessment of the algorithms' robustness against N_{td} is possible. For this purpose, the standard deviation of the averaged performance metrics for varying N_{td} are presented in Table 5.11. The corresponding database is archived in Section A.10.5. Analogously to the courses of Figure 5.20, the accuracy robustness of the proposed approaches is high (corresponds to a low standard deviation) in comparison to the comparative algorithm and vice versa in the case of precision accuracy. The spread of values for PH and MPAB indicates a high robustness for all approaches.

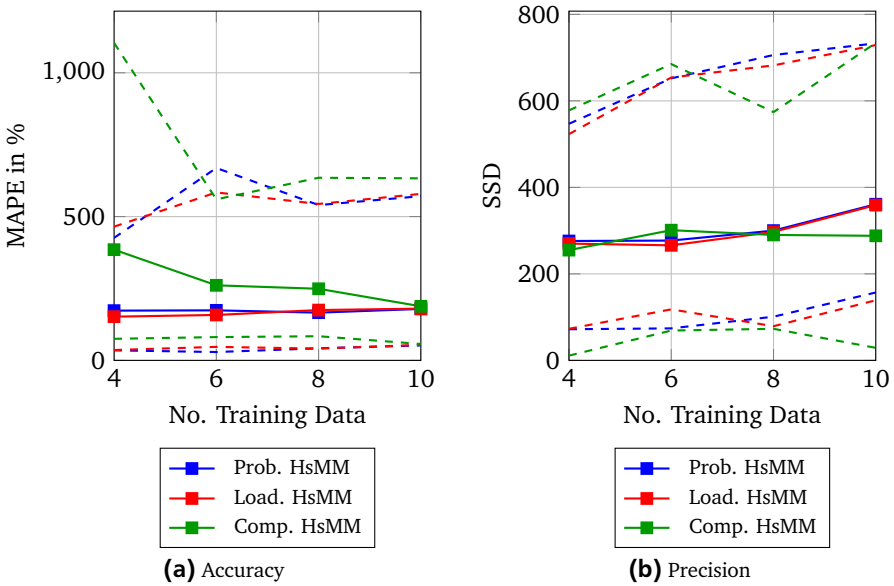


Figure 5.20.: Evolution of prognostic performance metrics for varying amount of training data (arithmetic mean (continuous line), 5th and 95th percentile (dashed lines))

Table 5.11.: Prognostic robustness based on averaged performance metrics with respect to varying number of training data N_{td}

Approach	$\sigma(\overline{\mathbf{PH}})$ [λ]	$\sigma(\overline{\mathbf{MAPE}})$ [%]	$\sigma(\overline{\mathbf{MPAB}})$ [%]	$\sigma(\overline{\mathbf{SSD}})$
Prob. HsMM	0.01	5	1.8	40
Load. HsMM	0.03	13	1.3	43
Comp. HsMM	0.01	82	2.6	20

5.5 Discussion and cost benefit analysis for current maintenance

This section completes the experimental analysis of the three examined HsMM approaches. Therefore, in the first subsection the achieved prognostic performance of each method is assessed and discussed. Furthermore, possible causes for the decreased performance in Section 5.4 are presented. This section concludes with a cost benefit assessment of the PHM algorithms in the context of aircraft maintenance.

5.5.1 Comparison of applied approaches

In this concluding subsection, the results of each HsMM approach achieved during verification and validations are analyzed. The first proposed HsMM approach is based on the duration probability in order to determine the sojourn time within a damage state during the prognostic step. Thus, the major advantage in comparison to the load-based HsMM approach is that measuring the cumulative load $\boldsymbol{u}_{\Sigma}$ as a second input is not necessary. This leads to benefits related to sensor suites and in the case of application fields, when the observed component is stressed by multiple kinds of loads. During the validation with all datasets, this approach showed a slightly better robustness with regard to a varying number of training data in comparison to the load-based method. However, in all other performance characteristics (accuracy, RUL uncertainty, precision), the load-based approach was slightly better than the probability-based concept. Especially during the validation with a selected subset in Section 5.3, the PH values were decreased in some cases. The overall comparison with all available datasets in Figure 5.19 indicated that the probability-based approach captures the smallest area of the radar chart, and thus showed the poorest performance in the majority of selected metrics. One possible explanation for this is the larger variety in state durations compared to the load-based approach: since the estimated duration times depend solely on probability distributions as depicted in Figure 5.11, the resulting spread of samples is higher, which can lead to reduced MAPE and PH values.

In contrast, the second HsMM approach, which considers the already endured load $\boldsymbol{u}_{\Sigma}$ for the RUL prediction, showed the best performance in nearly all trials. Especially during the overall validation in Section 5.4.1, the extension of the HsMM approach by one further input indicated benefits, since it outperformed the other methods in the majority of disciplines. Further investigations with regard to the impact of selected net probability $P(m_{net})$ revealed in Table 5.8 that a duration estimation with $P(m_{net}) = 1$ even slightly exceeds the values of SSD, PH, and MAPE of a likelihood-based model identification. This finding is not confirmed by consid-

ering the performance during the validation with a selected subset in Section 5.3.4, since the likelihood-based approach showed significantly better results compared to all other net probabilities. Besides the need for measuring the load, another drawback of this approach is the applied load-model, depicted as an example in Figure 5.13b. The first attempt in this thesis to interpolate and extrapolate the collected training data points is far from perfect. Further investigations focusing solely on the influences of cumulated load on the duration time within a damage state are recommended, especially with varying load amplitudes. Another disadvantage of this approach is the insignificant impact on some performance metrics such as the PH values in the context of validation with all datasets.

The comparative HsMM approach impresses by its simplicity, since no further effort for the identification of an underlying structure of damage states is necessary. This leads to a highly reduced process time for training and damage state estimation. However, the achieved prognostic performance is poorer compared to the results of the proposed HsMM approaches. In particular, the PH and MAPE values during the validation with all datasets indicated a lack in convergence and accuracy, respectively. One reason for the poor PH values is the wide prognostic uncertainty: as previously stated during the verification in Section 3.7, one outcome of assuming only four damage states within one training trial is the occurrence of multiple internal cycles in state transitions, which is part of the concept. However, this also leads to a wide spread of RUL samples during the prediction, so that the requirement of PH in Equation 2.6 for example is rarely fulfilled.

One further outcome of the comparison of all achieved prognostic performances are the strongly decreased metric values in the context of validation with all available datasets for the three examined HsMM approaches. Possible reasons for this are manifold. By inspecting the RMS of the vibration signal of all available datasets in Figure A.5 of Section A.7, one peculiarity is that only a few trials show a similar degradation process. However, the core assumption of a data-based PHM algorithm is similarity of trained and tested data in order to receive accurate RUL predictions. When neither of the trained models corresponds to the tested case, the predictions are unavoidably poor. Another reason could be the lack of monotony in the majority of runs. Although the consideration of more than one feature is assumed to reduce its impact (as proved in Section 5.3.5) by defining several damage states, cases occur, when all features return to their magnitudes at the beginning of the component's life cycle. The high spread of the failure distribution presented in Figure 5.3 also causes inaccuracies in spedific cases when the life cycle length of all trained runs strongly deviate from the tested run.

Further reasons for the disparity of degradations concern the test rig setup, since it offers many challenges to ensure comparability among different trials. The se-

lected bearing current as load for accelerated bearing aging has a strong stochastic impact on the degradation. As mentioned in Section 5.3.2, the state of the lubricant and other possible characteristics (initial contamination, manufacture of bearing components, etc.) influence the bearing current, which results in variations in damage progress. In addition, the attachment of the shaft coupling or the placement of the test bearing inside the hull could lead to different constellations of lever arm length, which could influence the angular deflection. This would have a direct effect on the frequency spectrum of the stator current, i.e. the features of the algorithms. Another explanation is connected to degradation of components apart from the tested bearing. For instance the hybrid bearing on the other side of the shaft could be degrading during the trials, since it is exposed to the vibrations induced by the test bearing. Although the functionality of the hybrid bearing was tested after every trial and its dimensioning was ensured, a subtle degradation during the trials is still possible. Another degrading component is the shaft which extends the motor shaft. As described in Section 5.1.3, it was necessary to replace this shaft due to corrosion; this exchange can also influence the angular deflection. Many other components can have similar impacts on the test rig, so that a complete comparability between different trials is difficult to ensure.

One indicator of the differences in degradation process among the trials is the prediction uncertainty. Thus, the averaged spread between the 5th and 95th percentile of all predictions $\overline{\Delta \mathrm{RUL}^*} = \mu(\mathrm{RUL}^*_{95} - \mathrm{RUL}^*_5)$ is evaluated for verification, and the two validations with a) selected subset and b) all datasets, respectively. In order to improve comparability, only the prediction uncertainty of the load-based HsMM approach is analyzed. To compensate differing length, the normed RUL is evaluated, and the results are presented in Table 5.12. Since the number of training datasets was set to 4 in verification as well as in validation with a subset, consequently only the results with four training datasets of the validation in Section 5.4 are applied. A trend can be derived among the cases, so that the spread of predicted RUL is largest in the case of validation with all datasets. Thus, in the case the prognostic algorithm assesses its own performance as being more uncertain in comparison to the verification and the first validation.

Table 5.12.: Normed prediction uncertainty in verification and validation with both selected subset and all datasets

	Verification	**Validation (subset)**	**Validation (all datasets)**
$\overline{\Delta \mathrm{RUL}^*}$	0.09	0.40	1.01

5.5.2 Monetary influences on current maintenance process

As introduced in Section 2.5, based on the results in [Käh17], possible cost savings of predictive maintenance (PM) in comparison to corrective maintenance can be estimated with respect to certain prognostic parameters. For this, the examined run to failure trials with degrading bearings must be embedded into a scenario of aircraft maintenance. Thus, several restrictions stem from this embedding:

- The examined component, which is not explicitly named in [Käh17] due to confidentiality, might differ from degrading bearings; however, bearings can fulfill the specified requirements of [Käh17, p. 42]: *correctively maintained*, *shows observable wear-out behavior*, *negative impact on aircraft operations*, etc. Thus, a bearing is assumed to have an analogous impact with regard to aircraft availability, repair time, and consequentially costs
- The tolerated failure threshold in aircraft maintenance is typically lower than the assumed failure level Ω. However, this can be easily adapted by defining a new failure state Ω (minimum state) which is far from actual breakdown
- Accelerated run to failure trials must be transformed into the life span of a real bearing to obtain important input parameters such as PH
- Besides the mean annual costs of the particular maintenance concept, the 2.5th and 97.5th percentile of expected PM costs are also determined in [Käh17, p. 137]; for a first guess, only the mean costs are applied in this thesis
- All underlying costs (indirect, direct maintenance costs, delay costs, etc.) and the PM implementation costs $C_P = 17{,}500$€/a (which is an input parameter of the simulation) are adapted from [Käh17]
- The total cost response surface graph in Figure 2.14 is a result of simulation; since the underlying simulation tool and corresponding process data is not available, the derived surface must be approximated analytically

The latter restriction leads to a qualitative approximation by [Käh17, p. 132]. Here, one significant point is the total avoidable cost of 283,000 €/a when there is no unexpected downtime of the aircraft. As mentioned in [Käh17, pp. 133–134], not all costs can be reduced, since the assumed PH is still too small to prevent all negative operational impacts. By including further boundaries (cost extrema for SFDR and FNR as 0 % or 100 %), the following approximation for cost savings ΔC,

i.e. current maintenance costs (corrective maintenance) subtracted from annual costs achieved with PM, is obtained:

$$\Delta C = 283 - 1.17 \cdot \mathrm{FNR} - 83 \cdot \exp(0.0299 \cdot \mathrm{SFDR}) \text{ [k€/a]} \tag{5.1}$$

Here, a PH of 300 min is assumed in [Käh17, p. 132]. This value must be related to the averaged achieved PH of $0.1\,\lambda$ in Table 5.8 for the proposed algorithms. Hence, two steps are necessary: transform the accelerated bearing degradation to the life span of bearings in real applications and convert the averaged PH values into minutes. In accordance with [SKF17], a bearing, which is loaded 8 h per day (e.g. gear drives or electric motors), is designed to last for between 10,000–25,000 h of operation. A bearing which is applied in aircraft systems can be assumed to be designed even more conservatively, i.e. with an increased life span. In the case a first incipient fault is recognized at half of the entire life cycle, the selected PH of 300 min would correspond to approximately 1 % of life span, so that in comparison to PH= $0.1\,\lambda$ (10 % of life span), it is a conservative assumption.

As depicted in Table 5.9, the resulting prediction performance expressed by SFDR and FNR varies only slightly among the three examined HsMM approaches. In Figure 5.21, the resulting cost benefits ΔC derived by the approximation of Equation 5.1 are plotted. In addition, the break-even threshold is sketched. The achieved prognostic performances are also marked as circles for the three HsMM approaches. The results indicate that the achieved performance of all three approaches lie on the left side of break-even, so that the application of PM with the proposed approaches is expected to be beneficial. However, due to the high error rates, all three approaches are near the break-even threshold.

The approximation of Equation 5.1 also offers the opportunity to quantify the results of Figure 5.21 in Table 5.13. As the distance between break-even line and achieved performances implies, the comparative approach creates the highest cost-benefit of 30,400 €/a (compared to 283,000 €/a avoidable costs). In comparison, both proposed HsMM approaches only generated a cost reduction of 15,600 €/a (probability-based approach) and 11,600 €/a (load-based approach). However, since the achieved PH of the comparative HsMM approach was only 0.05, the assumed PH=300 min of the scenario might not be met.

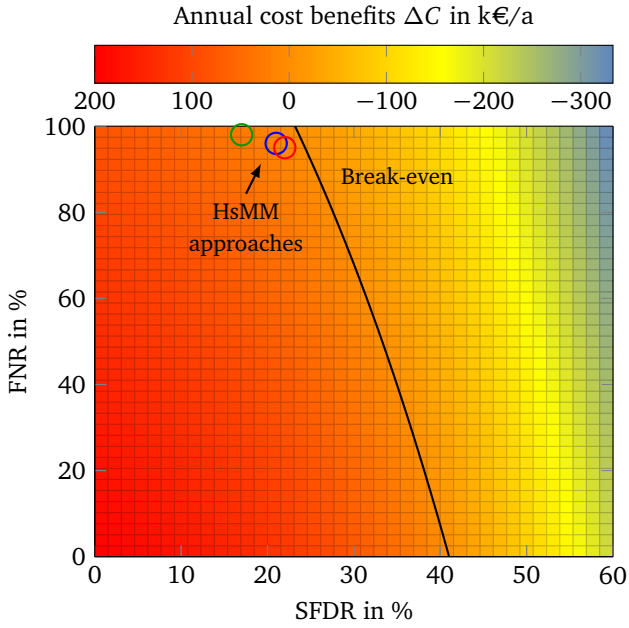


Figure 5.21.: Achieved cost benefits ΔC by application of probability-based HsMM (blue), load-based HsMM (red), and comparative HsMM (green) (PH= 300 min, $C_P = 17{,}500$ €/a). Based on [Käh17, p. 132]

Table 5.13.: Estimated savings for all three HsMM approaches based on approximation of Equation 5.1 and achieved SFDR and FNR values

	Prob. HsMM	**Load. HsMM**	**Comp. HsMM**
ΔC in k€/a	15.2	11.6	30.4

6 Summary and conclusions

This dissertation introduced a novel data-based algorithm for predictive maintenance (PM) of rotating components such as bearings or gear boxes. The core element of this algorithm is the Hidden semi-Markov Model (HsMM) approach for the identification (fault diagnosis) and propagation (fault prognosis) of the damage sequence of the examined component.

In the first chapter, the relevance of PM concerning safety, cost-efficiency, and sustainability was provided. Two main outcomes of this motivation were derived: the effective cost-benefits in case of PM introduction are still difficult to quantify, since the actual cost reductions can be annihilated due to the need for staff training and the immense upstream investment. Another challenge of PM is the consequences of the digitization: although the approach benefits from the availability of a large amount of data, it requires suitable data mining methods in order to separate significant data from irrelevant data. Thus, both aspects were examined within this thesis.

The focus of Chapter 2 was the integration of the outlined topic into the state-of-the-art terms and definitions of industrial maintenance. Besides common maintenance types (corrective, predetermined, and condition-based), the introduction of the Prognostics and Health Management (PHM) was central in this chapter. Here, the PHM comprises all steps for the support of PM: data acquisition, signal processing, fault diagnosis and fault prognosis, and advisory generation. Several state-of-the-art methods for each step were introduced. Data-based diagnosis and prognosis in particular were the focus of this survey. One approach in both fields is the application of HsMMs, which allows the modeling of sequences of dedicated damage states such as from *nominal state* via *first fault inceptions* to *failure state*. The outcome of this survey was that state-of-the-art HsMM approaches lack the identification of holistic damage states in different trials, so that each run is treated individually, since no interconnections between different runs are available.

In Chapter 3, this research gap was closed by the introduction of an HsMM that contains all possible damage sequences derived by the training datasets. By means of this so-called *net*, the merging of similar damage propagations is possible. In addition to the net, further HsMMs are trained with each particular dataset; with these single HsMMs it was ensured that test runs, which highly correlate with a training dataset, were reproduced very accurately. However, before the training of HsMMs, it was necessary to identify suitable features. Since the application field

of the proposed PHM algorithm was narrowed down to rotating components, one state-of-the-art method for feature generation is the transformation of the examined signals into the frequency domain in order to obtain more information about the current state of the component. The output is a multitude of features, which mostly contain no information or a similar amount of information. Thus, a feature reduction step is necessary in order to obtain only relevant features. For this, the *mutual information* criterion was selected to reduce the number of features and to simultaneously sustain the same amount of information. The resulting feature subset was then applied for HsMM training.

Based on the trained HsMMs, new test datasets are assessed in the context of fault diagnosis and prognosis. For this, all trained HsMMs are considered within a multiple model approach: as new data points are exploited, the probability of each HsMM (net and single) representing the underlying damage sequence most accurately is calculated. The derived model probability built the outputs of the fault diagnosis, in addition to the most probable current damage state. From here, two concepts in terms of fault prognosis were introduced in order to estimate the remaining useful life (RUL) of tested components. Both methods forecast the future duration time within each consecutive damage state until failure. The first approach was based solely on the duration probability distribution for every specific damage state derived during the training process. The second approach considered the already endured load for duration estimation. Both, the probability-based and load-based approaches were tested at the end of the chapter in terms of verification in order to confirm the mathematical correctness of the implemented algorithms. In addition, a state-of-the-art HsMM approach, which as previously mentioned lacks holistic damage states, was introduced and verified.

After the conceptual design of the proposed PHM algorithm, a case study was executed in Chapter 4. Based on trends in PHM, bearing degradation in induction machines was chosen to validate the suggested PHM algorithm. For this, a new test rig design and its realization were introduced in that chapter. For accelerated bearing aging, a so-called bearing current was applied as a load in order to significantly reduce the life cycle of bearings in the context of *run to failure* trials. Here, an alternating current runs through the test bearing; due to electrical discharge machining, all bearing components degrade rapidly.

The data derived from the test rig built the base for the analysis of the proposed PHM algorithm in Chapter 5. One novelty in comparison to state-of-the-art validations was the selected signal type. Instead of the widespread use of vibration data, the stator current of the induction machine was applied to identify the actual state of the bearing. Two different validations were executed: a leave one out cross validation with five representative datasets and a second validation with all available

datasets. The first validation indicated higher prognostic performances for the proposed algorithm for both applied duration estimation methods (probability-based and load-based HsMM approach) in comparison to the state-of-the-art HsMM algorithm. The load-based HsMM method in particular generated very accurate forecasts of unknown test datasets. A significant decrease in all evaluated performance metrics was detectable in the context of the second validation with all datasets. One identified potential reason was the discrepancy in degradation progresses within the examined datasets. Finally, one state-of-the-art tool for cost-benefit analysis of PHM algorithms in context of aircraft maintenance offered the opportunity to assess possible cost savings in the case of the introduction of the proposed HsMM approach. Despite the reduced prognostic performances of the second validation, the application can lead to savings, so that implementation is suggested. Further conclusions and a recommendation about future scientific work are presented in the following two sections.

6.1 Conclusions and recommendations

Several conclusions and recommendations are derived with regard to the tested algorithm and the applied test rig. These are summarized in the following itemization:

- The evaluation of PHM algorithms with stator current signals instead of vibration signals is possible; in terms of a validation (with selected subset of data), the predictions with examined stator current were even more accurate and precise than with vibration signals (e.g. $\overline{\mathrm{PH}}_{vib} = 0.19\,\lambda$, $\overline{\mathrm{PH}}_{curr} = 0.51\,\lambda$)

- Consideration of the endured load for fault prognosis led to better prognostic performances in nearly all examinations (e.g. $\overline{\mathrm{PH}}_{load} = 0.51\,\lambda$ $\overline{\mathrm{PH}}_{prob} = 0.18\,\lambda$ during validation with data subset); thus, an extension of the applied input vector for the estimation of RUL was beneficial in the context of the employed test procedure (load cycles, load spectrum, etc.).

- The generation of holistic damage states, which is the main novelty of this thesis, resulted in more accurate, precise, and robust performance compared to state-of-the-art HsMM approaches (e.g. $\overline{\mathrm{MAPE}}_{load} = 166\,\%$, $\overline{\mathrm{MAPE}}_{comp} = 271\,\%$).

- The expected cost-benefits of the suggested scenario with the achieved prognostic performances varies between 11.6 k€/a to 15.2 k€/a (in comparison to 283 k€/a avoidable maintenance costs) for the proposed algorithm.

- The applied bearing current as a load for accelerated bearing aging provided rapid degradations (10 h – 150 h); however, the resultant degradation processes varied greatly among the trials. A change in the applied test procedure is therefore recommended: lower voltage of bearing current (instead of 10 V) and continuous load instead of load cycles (better monotony is assumed).

- The combination of selected bearing type (with polyamide cage) and applied load resulted in less abrupt failures in comparison to bearings with steel cage.

6.2 Outlook

One of the most crucial aspects during the implementation of the presented algorithm was the selection of suitable features that represent the actual degradation. Although the applied selection criterion based on mutual information and standard deviation of each feature provided a reasonable feature set, several improvements are possible. One example is that currently, the number of features must be selected by the user. A superior information criterion, which evaluates the total information content of a selected feature subset, could support or automate this parameterization step.

Whereas this dissertation focused on the evaluation of motor current signals so that the measured vibration signal was applied only as a comparative signal, the consideration of both signal types could increase the prognostic performance. A mixture of both signals in the applied feature matrix could be a simple implementation. The derived research question would target the actual increase in prognostic performance compared to implementation and product costs.

One last recommendation for future scientific work is related to the embedding of load into the RUL forecast. Since this consideration resulted in increased prognostic performance, further research effort in this field is suggested. This includes a change in the applied load spectrum (randomly selected load levels and load cycle lengths, combination of different load types, etc.). However, also the influences of load on estimated damage states or the duration time within these states could also be the focus of further research. This could include for example the extension of the selected load-model by more advanced regression methods in the field of machine learning, e.g. Gaussian Process Regression or Relevance Vector Machines.

Bibliography

[ABS+13] Patrick Adam, Anton Borger, Markus Schwind, Hechun Shen, and Zhenyin Wang. *Konzeptionierung eines Prüfstandes zur Generierung von Lagerschadendaten für den Bereich der zustandsbasierten Wartung*. ADP Final report, Technische Universität Darmstadt, Darmstadt, September 2013.

[Agg15] Charu C. Aggarwal. *Data mining: The textbook*. Springer, Cham [Switzerland] and New York, [paperback edition] edition, 2015.

[Alp10] Ethem Alpaydin. *Introduction to machine learning*. MIT Press, Cambridge (Mass.), 2. ed. edition, 2010.

[AM01] J. Altmann and J. Mathew. Multiple Band-pass Autoregressive Demodulation for Rolling-element Bearing Fault Diagnosis. *Mechanical Systems and Signal Processing*, 15(5):963–977, 2001.

[ANMK06] T. Akagaki, M. Nakamura, T. Monzen, and M. Kawabata. Analysis of the Behaviour of Rolling Bearings in Contaminated Oil Using Some Condition Monitoring Techniques. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 220(5):447–453, 2006.

[Ant07] J. Antoni. Cyclic spectral analysis of rolling-element bearing signals: Facts and fictions. *Journal of Sound and Vibration*, 304(3-5):497–529, 2007.

[APK15] Christoph Anger, Christian Preusche, and Uwe Klingauf. Asynchronous motor test bench for the generation and current signal diagnostics of accelerated bearing damage. *Proceedings of Annual PHM Society Conference*, 2015(6), 2015.

[Bau08] Alexander Baumeister. *Lebenszykluskosten alternativer Verfügbarkeitsgarantien im Anlagenbau*, volume Bd. 117 of *Gabler Edition Wissenschaft*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler / GWV Fachverlage, Wiesbaden, Wiesbaden, 1. aufl. edition, 2008.

[BBV14] Edward Max Bertot, Pierre-Philippe Beaujean, and David Vendittis. Refining Envelope Analysis Methods using Wavelet De-Noising to Identify Bearing Faults. *Proceedings of Second European Conference of the PHM Society 2014*, 2:119–126, 2014.

[BCFP14] Zhang Bingzhi, Ding Chuang, Feng Fuzhou, and Jiang Pengcheng. Development of testable shaft for equipment PHM system. In *Proceedings of 2014 Prognostics and System Health Management Conference (PHM-2014 Hunan)*, pages 6–10, [Piscataway, N.J.], 2014. IEEE.

[BDGM14] Jean-Paul Beetz, Steffen Dressler, Samira Gruber, and Thomas Meißner. *Weiterentwicklung des bestehenden Konzepts eines Prüfstandes zur Generierung von Lagerschadendaten*. ADP Final report, Technische Universtät Darmstadt, Darmstadt, 04.03.2014.

[Bee04] Raymond S. Beebe. *Predictive maintenance of pumps using condition monitoring*. Elsevier Advanced Technology, Kidlington, Oxford, UK and New York, 2004.

[Bel08] Bellini, Alberto and Immovilli, Fabio and Rubini, Ricardo and Tassoni, Carla. Diagnosis of Bearing Faults in Induction Machines by Vibration or Current

Signals: A Critical Comparison. *Industry Applications Society Annual Meeting, 2008. IAS'08. IEEE*, 46(4):1–8, 2008.

[BGRR08] M. Blödt, P. Granjon, B. Raison, and G. Rostaing. Models for Bearing Damage Detection in Induction Motors Using Stator Current Monitoring. *IEEE Transactions on Industrial Electronics*, 55(4):1813–1822, 2008.

[BH02] Hugh E. Boyanton and Gary Hodges. Bearing fluting: the results of a long-term investigation into bearing fluting on AC motors, DC motors, and Rolls on paper machines. *IEEE Industry Applications Magazine*, 2002(Sep/Oct issue):53–57, 2002.

[Bis13] Christopher M. Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York [u.a.], 11. (corr. printing) edition, 2013.

[Blo84] Henk A. P. Blom. An efficient filter for abruptly changing systems. In *The 23rd IEEE Conference on Decision and Control*, pages 656–658. IEEE, 1984.

[BP66] Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

[BSGR14] Anthony Barré, Frédéric Suard, Mathias Gérard, and Delphine Riu. A Real-time Data-driven Method for Battery Health Prognostics in Electric Vehicle Use. *Proceedings of Second European Conference of the PHM Society 2014*, pages 15–22, 2014.

[Bun14] Bundesministerium für Wirtschaft und Energie. Die Luftfahrtstrategie der Bundesregierung, March 2014.

[Bun16] Bundesministerium für Wirtschaft und Energie. Bekanntmachung zur Förderung von Forschungs- und Technologievorhaben im Rahmen des fünften nationalen zivilen Luftfahrtforschungsprogramms Dritter Programmaufruf, 01.09.2016.

[Cal09] Keith Calhoun. Health Management at Rolls-Royce. `https://www.phmsociety.org/sites/phmsociety.org/files/FieldedSystems_Calhoun.pdf`, 2009.

[CP04] Shuang Cang and Derek Partridge. Feature ranking and best feature subset using mutual information. *Neural Computing and Applications*, 13(3):175–184, 2004.

[Deu10] Deutsches Institut für Normung e.V. Begriffe der Instandhaltung – DIN EN 13306, 2010.

[Deu12] Deutsches Institut für Normung e.V. Grundlagen der Instandhaltung – DIN 31051, 2012.

[DGO11] M. Delgado, A. Garcia, and J. A. Ortega. Evaluation of feature calculation methods for electromechanical system diagnosis. In *SDEMPED 2011*, pages 495–502, Piscataway, NJ, 2011. IEEE.

[DQQG15] Dong Wang, Qiang Miao, Qinghua Zhou, and Guangwu Zhou. An Intelligent Prognostic System for Gear Performance Degradation Assessment and Remaining Useful Life Estimation. *Journal of Vibration and Acoustics*, 2015.

[DSG12] Matthew Daigle, Bhaskar Saha, and Kai Goebel. A comparison of filter-based approaches for model-based prognostics. In *Aerospace Conference, 2012 IEEE*, pages 1–10, 2012.

[Dup10] Richard Dupuis. Application of oil debris monitoring for wind turbine gearbox prognostics and health management. In *Annual Conference of the prognostics and health management society*, page 10, 2010.

[DV07] J. Dzakowic and G. S. Valentine. Advanced Techniques for the Verification and Validation of Prognostics & Health Management Capabilities. *Machinery Failure Prevention Technologies*, 60, 2007.

[ECBC12] El Bouchikhi El Houssin, Vincent Choqueuse, Mohamed Benbouzid, and Jean Frédéric Charpentier. *IEEE International Symposium on Industrial Electronics (ISIE), 2012: 28 - 31 May 2012, Hangzhou Tianyuan Tower Hotel, Hangzhou, China ; proceedings*. IEEE, Piscataway, NJ, 2012.

[ED04] L. Eren and M.J Devaney. Bearing Damage Detection via Wavelet Packet Decomposition of the Stator Current. *IEEE Transactions on Instrumentation and Measurement*, 53(2):431–436, 2004.

[EM10] M. Elforjani and D. Mba. Accelerated natural fault diagnosis in slow speed bearings with Acoustic Emission. *Engineering Fracture Mechanics*, 77(1):112–127, 2010.

[Eur11] European Commission. *Flightpath 2050: Europe's vision for aviation ; maintaining global leadership and serving society's needs ; report of the High-Level Group on Aviation Research*. Policy / European Commission. Publ. Off. of the Europ. Union, Luxembourg, 1 edition, 2011.

[Fer80] Jack D. Ferguson. Variable duration models for speech. In *Proceedings of the Symposium on the Application of HMMs to Text and Speech*, pages 143–179, 1980.

[Fie15] Andy Field. *Discovering statistics using IBM SPSS statistics: And sex and drugs and rock'n'roll*. SAGE, Los Angeles, 4th ed., reprinted. edition, 2015.

[GCLR14] Grant Galloway, Victoria Catterson, Craig Love, and Andrew Robb. Anomaly detection techniques for the condition monitoring of tidal turbines. In *Annual Conference of the Prognostics and Health Management Society 2014 (PHM)*, 2014.

[GS04] Paresh Girdhar and C. Scheffer. *Practical machinery vibration analysis and predictive maintenance*. Practical professional books from Elsevier. Elsevier and Newnes, Amsterdam and Boston and Burlington, MA, 2004.

[Han10] L. Han. Bearing Fault Detection Based on Order Bispectrum. *Image and Signal Processing (CISP)*, 3(7):3405–3408, 2010.

[Har03] J. Harrison. Formal verification at Intel. In *18th Annual IEEE Symposium on Logic in Computer Science*, pages 45–54, Los Alamitos, Calif., 2003. IEEE Computer Society Press.

[HE96] David Kulp David Haussler and Eeckman, Martin G Reese Frank H. A generalized hidden Markov model for the recognition of human genes in DNA. In *Proc. Int. Conf. on Intelligent Systems for Molecular Biology, St. Louis*, pages 134–142, 1996.

[Hei03] Jens Heim. Wälzlager mit Sensoren: German patent No. DE000010215929A1, 2003.

[Hun96] T.M Hunt. *Condition monitoring of mechanical and hydraulic plant: A concise introduction and guide*. Chapman & Hall, London, 1996.

[HW14] Rüdiger Hoffmann and Matthias Wolff. *Signalanalyse*, volume 1 of *Intelligente Signalverarbeitung / Rüdiger Hoffmann*. Springer Vieweg, Berlin [u.a.], 2. aufl. edition, 2014.

[IAT15] IATA. Airline Maintenance Cost Executive Commentary, 2015.

[ICBR09] F. Immovilli, M. Cocconcelli, A. Bellini, and R. Rubini. Detection of Generalized-Roughness Bearing Fault by Spectral-Kurtosis Energy of Vibration or Current Signals. *IEEE Transactions on Industrial Electronics*,

56(11):4710–4717, 2009.

[Int03] International Organization for Standardization. Condition monitoring and diagnostics of machines — Data processing, communication and presentation – 13374-1, 15.03.2003.

[Ise06] Rolf Isermann. *Fault-diagnosis systems: An introduction from fault detection to fault tolerance*. Springer, Berlin and New York, op. 2006.

[JA05] José Silva and A.J. Marques Cardoso. Bearing Failures Diagnosis in Three-Phase Induction Motors by Extended Park's Vector Approach. *IECON 2005. 31st Annual Conference of IEEE*, 31:2591–2596, 2005.

[JCD+15] Huiming Jiang, Jin Chen, Guangming Dong, Tao Liu, and Gang Chen. Study on Hankel matrix-based SVD and its application in rolling element bearing fault diagnosis. *Mechanical Systems and Signal Processing*, 52-53:338–359, 2015.

[Jen13] Ian K. Jennions. *Integrated vehicle health management: The technology*. Society of Automotive Engineers. Electronic publications. Society of Automotive Engineers, Warrendale, Pa. (400 Commonwealth Dr., Wallendale PA USA), 2013.

[JMC+09] Jeremy S. Sheldon, Matthew J. Watson, Carl S. Byington, Genna M. Swerdon, and and M. Begin. Integrating Model-based Shaft Coupling Prognostics with Vibration Diagnostic Features. *Australian International Aerospace Congress*, 13, 2009.

[JOL08] S. Janjarasjitt, H. Ocak, and K.A Loparo. Bearing condition diagnosis and prognosis using applied nonlinear dynamical analysis of machine vibration signal. *Journal of Sound and Vibration*, 317(1-2):112–126, 2008.

[Kad13] Seifedine Kadry. *Diagnostics and prognostics of engineering systems: Methods and techniques*. Engineering Science Reference, Hershey Pa., 2013.

[Käh17] Alexander Kählert. *Specification and Evaluation of Prediction Concepts in Aircraft Maintenance*. PhD thesis, Technische Universtät Darmstadt, Darmstadt, 2017.

[KAS12] J. Karandikar, A. Abbas, and T. Schmitz. Remaining useful tool life predictions in turning using Bayesian inference. *International Journal of Prognostics and health Management*, 2012.

[KBA08] K. Goebel, B. Saha, and A. Saxena. A Comparison of Three Data-driven Techniques for Prognostics. *62nd Meeting of the Society For Machinery Failure Prevention Technology (MFPT)*, pages 119–131, 2008.

[KCT+14] Buyung Kosasih, Wahyu Caesarendra, Kiet Tieu, Achmad Widodo, Craig A.S. Moodie, and A. Kiet Tieu. Degradation Trend Estimation and Prognosis of Large Low Speed Slewing Bearing Lifetime. *Applied Mechanics and Materials*, 493:343–348, 2014.

[Kli90] G. B. Kliman. Induction Motor Fault Detection via Passive Current Monitoring: A brief Survey. *Proceedings of the 44th Meeting of the Mechanical Failures Prevention Group*, 44:49–66, 1990.

[KP02] Kim and Parlos. Induction motor fault diagnosis based on neuropredictors and wavelet signal processing - Mechatronics, IEEE/ASME Transactions on. *IEEE Transactions on Industrial Electronics*, 2002.

[LBC15] T. T. LE, C. Berenguer, and F. Chatelain. Prognosis based on Multi-branch Hidden semi-Markov Models: A case study. *IFAC-PapersOnLine*, 48(21):91–96, 2015.

[LDP12] Qinming Liu, Ming Dong, and Ying Peng. A novel method for online health prognosis of equipment based on hidden semi-Markov model using sequential Monte Carlo methods. *Mechanical Systems and Signal Processing*, 32:331–348, 2012.

[Le15] Thanh Trung Le. *Contribution to deterioration modeling and residual life estimation based on condition monitoring data*. PhD thesis, Université Grenoble Alpes, 01.01.2015.

[Lei14] Bernhard Leidinger. *Wertorientierte Instandhaltung*. Springer Fachmedien Wiesbaden, Wiesbaden, 2014.

[LSGB15] Xin Lei, Peter A. Sandborn, Navid Goudarzi, and Maira A. Bruck. PHM based predictive maintenance option model for offshore wind farm O&M optimization. *Annual Conference of the Prognostics and Health Management Society, Coronado Island Marriott in Coronado, California*, 2015.

[LWZ+14] Jay Lee, Fangji Wu, Wenyu Zhao, Masoud Ghaffari, Linxia Liao, and David Siegel. Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. *Mechanical Systems and Signal Processing*, 42(1-2):314–334, 2014.

[Mat10] Kurt Matyas. *Taschenbuch Instandhaltungslogistik*. Carl Hanser Verlag GmbH & Co. KG, München, 2010.

[Mik15] Heiko Mikat. *Hybride Fehlerprognose zur Unterstützung prädiktiver Instandhaltungskonzepte in der Luftfahrt*. Dissertation, TU Darmstadt, Darmstadt, 2015.

[MIM01] MIMOSA. Open System Architecture for Condition-Based Maintenance. `http://www.mimosa.org/mimosa-osa-cbm`, 2001.

[MK06] A. R. Mohanty and Chinmaya Kar. Fault detection in a multistage gearbox by demodulation of motor current waveform. *Industrial Electronics, IEEE Transactions on*, 53(4):1285–1297, 2006.

[MK07] Mark Schwabacher and Kai Goebel. A Survey of Artificial Intelligence for Prognostics. *AAAI fall symposium*, pages 107–114, 2007.

[Mob02] R. Keith Mobley. *An introduction to predictive maintenance*. Butterworth-Heinemann, Amsterdam and New York, 2nd ed. edition, 2002.

[MPP07] J. F. Martins, V. F. Pires, and A. J. Pires. Unsupervised Neural-Network-Based Algorithm for an On-Line Diagnosis of Three-Phase Induction Motor Stator Fault. *IEEE Transactions on Industrial Electronics*, 54(1):259–264, 2007.

[MS85] P. D. McFadden and J. D. Smith. The vibration produced by multiple point defects in a rolling element bearing. *Journal of Sound and Vibration*, 98(2):263–273, 1985.

[MTMZ12] Kamal Medjaher, Diego Alejandro Tobon-Mejia, and Noureddine Zerhouni. Remaining Useful Life Estimation of Critical Components With Application to Bearings. *IEEE Transactions on Reliability*, 61(2):292–302, 2012.

[Mur02] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. Dissertation, UNIVERSITY OF CALIFORNIA, BERKELEY, 2002.

[Nad04] Robert B. Randall Nader Sawalhi. The Application of Spectral Kurtosis to Bearing Diagnostics. *Proceedings of Acoustics 2004*, 2004.

[NAP10] Gang Niu, Davinder Anand, and Michael Pecht. Prognostics and health management for energetic material systems. In *IEEE Prognostics & System Health Management Conference 2010*, pages 1–7, [Piscataway, N.J.], 2010. IEEE.

[NSS96] Krishna Nathan, Andrew Senior, and Jayashree Subrahmonia. Initialization of Hidden Markov Models for Unconstrained On-line Handwriting Recog-

nition. *IBMIEEE Conference on Audio Speech and Signal Processing*, pages 3482–3485, 1996.

[OCH+14] Didem Ozevin, Justin Cox, William Hardman, Seth Kessler, and Alan Timmons. Fatigue Crack Detection at Gearbox Spline Component using Acoustic Emission Method. *Annual Conference of the Prognostics and Health Management Society 2014*, 2014.

[OLD07] Hasan Ocak, Kenneth A. Loparo, and Fred M. Discenzo. Online tracking of bearing wear using wavelet packet decomposition and probabilistic modeling: A method for bearing prognostics. *Journal of Sound and Vibration*, 302(4-5):951–961, 2007.

[PAK14] Christian Preusche, Christoph Anger, and Uwe Klingauf. Evaluation of the Training Process of three different Prognostic Approaches based on the Gaussian Process. *Proceedings of Second European Conference of the PHM Society 2014*, 2, 2014.

[Pal59] Arvid Palmgren. Ball and roller bearing engineering. *Philadelphia: SKF Industries Inc., 1959*, 1, 1959.

[Pap17] Christoph Papenfuss. Ist prädiktive Instandhaltung die Killer-App für das Industrial Internet of Things? *Industrie 4.0 Management*, 2017(33):57–60, 2017.

[PBG12] Ashok Prajapati, James Bechtel, and Subramaniam Ganesan. Condition based maintenance: A survey. *Journal of Quality in Maintenance Engineering*, 18(4):384–400, 2012.

[PDN+01] M. Pecht, M. Dube, M. Natishan, R. Williams, J. Banner, and I. Knowles. Evaluation of built-in test. *IEEE Transactions on Aerospace and Electronic Systems*, 37(1):266–271, 2001.

[PJ09] M. Pecht and Jie Gu. Physics-of-failure-based prognostics for electronic products. *Transactions of the Institute of Measurement and Control*, 31(3-4):309–322, 2009.

[PN06] H. Powrie and A. Novis. Gas Path Debris Monitoring for F-35 Joint Strike Fighter Propulsion System PHM. In *IEEE Aerospace Conference, 2006*, pages 1–8, Piscataway, NJ, 2006. IEEE Operations Center.

[Pow85a] Power Systems Reliability Subcommittee of the IEEE Industry Applications Society. Report of Large Motor Reliability Survey of Industrial and Commercial Installations, Part I. *IEEE Transactions on Industry Applications*, IA-21(4):853–864, 1985.

[Pow85b] Power Systems Reliability Subcommittee of the IEEE Industry Applications Society. Report of Large Motor Reliability Survey of Industrial and Commercial Installations, Part II. *IEEE Transactions on Industry Applications*, IA-21(4):865–872, 1985.

[PPG+13] Bharath Pattipati, Krishna Pattipati, Youssef A. Ghoneim, Mark Howell, and Mutasim A. Salman. Electronic Returnless Fuel System Fault Diagnosis and Isolation: a Data-driven Approach. *Annual Conference of the Prognostics and Health Management Society 2013*, 4(4):57–65, 2013.

[Rab89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[RAC01] R. B. Randall, J. Antoni, and S. Chobsaard. The Relationship between Spectral Correlation and Envelope Analysis in the Diagnostics of Bearing Faults and other Cyclostationary Machine Signals. *Mechanical Systems and Signal Processing*, 15(5):945–962, 2001.

[Rou87] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[RRBM02] B. Raison, G. Rostaing, O. Butscher, and C.-S Maroni. Investigations of algorithms for bearing fault detection in induction drives. *IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the]*, 2(1):1696–1701, 2002.

[RS16] R. B. Randall and Wade Smith. New cepstral methods for the diagnosis of gear and bearing faults under variable speed conditions. *International Congress on Sound & Vibration*, 23, 2016.

[SB10] John A. Sokolowski and Catherine M. Banks. *Modeling and Simulation Fundamentals*. John Wiley & Sons, Inc, Hoboken, NJ, USA, 2010.

[SCB+08] Abhinav Saxena, Jose Celaya, Edward Balaban, Kai Goebel, Bhaskar Saha, Sankalita Saha, and Mark Schwabacher. Metrics for evaluating performance of prognostic techniques. *Prognostics and health management, 2008. phm 2008. international conference IEEE*, pages 1–17, 2008.

[Sch78] Gideon Schwarz. Estimating the Dimension of a Model. *The annals of statistics*, 6(6):461–464, 1978.

[Sch10] Jaap Schijve, editor. *Fatigue of structures and materials*. Springer, Amsterdam, 2. ed. edition, 2010.

[Sch11] Jochen Schaab. *Trusted health assessment of dynamic systems based on hybrid joint estimation*, volume Nr. 1181 of *Fortschritt-Berichte VDI. Reihe 8, Mess-, Steuerungs- und Regelungstechnik*. VDI, Düsseldorf, 2011.

[Sch15] Tim Schickel. *Inbetriebnahme eines Prüfstandes zur Generierung von Lagerschadendaten und Validierung eines low-cost Sensorsystems: Commissioning of a Test Bed for the Generation of Bearing Fault Data and for the Validation of a low-cost Sensor System*. Master's thesis, Technische Universität Darmstadt, Darmstadt, January 2015.

[SCR13] Abdenour Soualhi, Guy Clerc, and Hubert Razik. Detection and Diagnosis of Faults in Induction Motor Using an Improved Artificial Ant Clustering Technique. *IEEE Transactions on Industrial Electronics*, 60(9):4053–4062, 2013.

[SCS+10] Abhinav Saxena, Jose Celaya, Bhaskar Saha, Sankalita Saha, and Kai Goebel. Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and health Management*, 1(1):4–23, 2010.

[SCWG09] Bhaskar Saha, Jose R. Celaya, Philip F. Wysocki, and Kai F. Goebel. Towards prognostics for electronics components. In *Aerospace conference, 2009 IEEE*, pages 1–7, 2009.

[SGC09] Bhaskar Saha, Kai Goebel, and Jon Christophersen. Comparison of prognostic algorithms for estimating remaining useful life of batteries. *Transactions of the Institute of Measurement and Control*, 2009.

[SHH04a] J. R. Stack, T. G. Habetler, and R. G. Harley. Fault Classification and Fault Signature Production for Rolling Element Bearings in Electric Machines. *IEEE Transactions on Industry Applications*, 40(3):735–739, 2004.

[SHH04b] J.R Stack, T.G Habetler, and R.G Harley. Bearing Fault Detection via Autoregressive Stator Current Modeling. *IEEE Transactions on Industry Applications*, 40(3):740–747, 2004.

[SHH05] J. R. Stack, T. G. Habetler, and R. G. Harley. Experimentally Generating Faults in Rolling Element Bearings Via Shaft Current. *IEEE Transactions on Industry Applications*, 41(1):25–29, 2005.

[SHKB95] R. R. Schoen, T. G. Habetler, F. Kamran, and R. G. Bartfield. Motor bearing damage detection using stator current monitoring. *IEEE Transactions on Industry Applications*, 31(6):1274–1279, 1995.

[SKF17] SKF. Bearing life and load ratings. `http://www.skf.com/pk/products/bearings-units-housings/roller-bearings/principles/selecting-bearing-size/bearing-life/index.html`, 2017.

[SLH+95] R. R. Schoen, B. Lin, T.G Habetler, J. H. Schlag, and S. Farag. An Unsupervised, On-Line System for Induction Motor Fault Detection Using Stator Current Monitoring. *IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS,*, 6(31):1280–1286, 1995.

[SLNL14] Manjeevan Seera, Chee Peng Lim, Saeid Nahavandi, and Chu Kiong Loo. Condition monitoring of induction motors: a review and an application of an ensemble of hybrid intelligent models. *Expert Systems with Applications*, 41(10):4891–4903, 2014.

[SR03] Alexander K. Schömig and Oliver Rose. On the suitability of the Weibull distribution for the approximation of machine failures. In *IIE Annual Conference. Proceedings*, pages 1–7, 2003.

[SSSG09] Abhinav Saxena, Bhaskar Saha, Sankalita Saha, and Kai Goebel. Evaluating algorithm performance metrics tailored for prognostics. *2009 IEEE Aerospace conference*, pages 1–13, 2009.

[Str12] Matthias Strunz. *Instandhaltung: Grundlagen, Strategien, Werkstätten*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[TL14] Tiedo Tinga and Richard Loendersloot. Aligning PHM, SHM and CBM by understanding the physical system failure behaviour. *Proceedings of Second European Conference of the PHM Society 2014*, pages 162–171, 2014.

[TMMZT12] Diego Alejandro Tobon-Mejia, Kamal Medjaher, Noureddine Zerhouni, and Gerard Tripot. A Data-Driven Failure Prognostics Method Based on Mixture of Gaussians Hidden Markov Models. *IEEE Transactions on Reliability*, 61(2):491–503, 2012.

[TSS06] T. R. Kurfess, S. Billington, and S. Y. Liang. Advanced Diagnostic and Prognostic Techniques for Rolling Element Bearings. *Condition Monitoring and Control for Intelligent Manufacturing*, pages 137–165, 2006.

[TvFM01] P. Tappert, A. von Flotow, and M. Mercadal. Autonomous PHM with blade-tip-sensors: algorithms and seeded fault experience. In *Aerospace Conference, 2001, IEEE Proceedings*, pages 7–3295. IEEE, 2001.

[U.S10] U.S. Department of Energy. Operations & Maintenance Best Practices: A Guide to Achieving Operational Efficiency, 2010.

[Ver99] Verein Deutscher Ingenieure. Zustandsorientierte Instandhaltung, Dezember 1999.

[VG15] George J. Vachtsevanos and Kai Goebel. Tutorial on Introduction to Prognostics, Annual Conference of the PHM Society 2015. `https://www.phmsociety.org/sites/phmsociety.org/files/PROGNOSTICS_TUTORIAL.pdf`, 2015.

[VLR+06] George J. Vachtsevanos, Frank L. Lewis, Michael Roemer, Andrew Hess, and Biqing Wu. *Intelligent fault diagnosis and prognosis for engineering systems*. Wiley, Hoboken N.J., 2006.

[VP06] N. M. Vichare and M. G. Pecht. Prognostics and health management of electronics. *IEEE Transactions on Components and Packaging Technologies*, 29(1):222–229, 2006.

[VRK03] Venkat Venkatasubramanian, Raghunathan Rengaswamy, and Surya N. Kavuri. A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies. *Computers & Chemical Engineering*, 27(3):313–326, 2003.

[VRKY03] Venkat Venkatasubramanian, Raghunathan Rengaswamy, Surya N. Kavuri, and Kewen Yin. A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & Chemical Engineering*, 27(3):327–346, 2003.

[VRYK03] Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewen Yin, and Surya N. Kavuri. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, 27(3):293–311, 2003.

[WC05] Jian-Da Wu and Chao-Qin Chuang. Fault diagnosis of internal combustion engines using visual dot patterns of acoustic and vibration signals. *NDT & E International*, 38(8):605–614, 2005.

[WMZZ15] Dong Wang, Qiang Miao, Qinghua Zhou, and Guangwu Zhou. An Intelligent Prognostic System for Gear Performance Degradation Assessment and Remaining Useful Life Estimation. *Journal of Vibration and Acoustics*, 137(2):1–12, 2015.

[WNS+14] Zachary Welz, Alan Nam, Michael Sharp, J. Wesley Hines, and Belle R. Upadhyaya. Prognostics for Light Water Reactor Sustainability: Empirical Methods for Heat Exchanger Prognostic Lifetime Predictions. In *2nd European Conference of the Prognostics and Health Management Society (PHME'14), Nantes, France, July*, pages 8–10, 2014.

[YGA11] C. T. Yiakopoulos, K. C. Gryllias, and I. A. Antoniadis. Rolling element bearing fault detection in industrial environments based on a K-means clustering approach. *Expert Systems with Applications*, 38(3):2888–2911, 2011.

[YK06] Shun-Zheng Yu and H. Kobayashi. Practical implementation of an efficient forward-backward algorithm for an explicit-duration hidden Markov model. *IEEE Transactions on Signal Processing*, 54(5):1947–1951, 2006.

[YOI92] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden Markov model. In *Computer Vision and Pattern Recognition (CVPR '92)*, pages 379–385, Los Alamitos, Oct. 1992. IEEE Computer Society Press.

[YQI08] Weizhong Yan, Hai Qiu, and Naresh Iyer. Feature extraction for bearing prognostics and health management: A survey. *Meeting of the Society for Machinery*, 62, 2008.

[ZP07] Jafar Zarei and Javad Poshtan. Bearing fault detection using wavelet packet transform of induction motor stator current. *Tribology International*, 40(5):763–769, 2007.

A Appendix

This chapter offers more information on several aspects of the proposed algorithm, the applied data, and validation. Thus, it supports the main part of this thesis by additional analyses, auxiliary plots, and pseudo code of the applied algorithms.

A.1 Evaluation of different classification algorithms

As presented in Section 3.3.1, several classification algorithms that are available in the *classification learner application* of Matlab were tested. The algorithms had to classify the results of the clustering by K-means as targets in accordance to the feature matrix as inputs. The test included a holdout validation so that 50 % of the data was used for training and the other 50 % of the data points was applied for the evaluation. Three different kinds of classifiers were evaluated: Decision trees with different numbers of splits, SVMs with several kernels, and KNN with a varying number of neighbors. The results are summarized in Table A.1. One extraction of the confusion matrix of an inaccurate classification by a simple decision tree is presented in Figure A.1.

Table A.1.: Evaluation results of different classification algorithm

Algorithm	**Decision Tree**			**SVM**			**KNN**		
Parameter	Number of splits			Kernel			Number of neighbors		
	4	20	100	Linear	Cubic	Gauss	1	5	6
Accuracy [%]	53	80	100	100	100	99	100	100	99

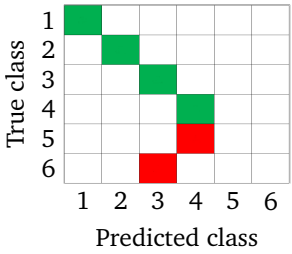


Figure A.1.: Extraction of a confusion matrix to display the misclassification of Simple Tree approach (four splits)

A.2 Basics of the Hidden semi-Markov Model

In this section, the mathematical formulations for the applied Forward-backward approach to train Hidden semi-Markov Models (HsMM) are presented. The formulations and notations base on the implementations introduced in [YK06]. The entire algorithm is divided into the Forward-backward step in Table A.2 and the subsequent parameter re-estimation step in Table A.3. Both tables illustrate the algorithms as pseudo code.

The inputs of the algorithm are initial values for the required parameters $\boldsymbol{\lambda} = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{D}, \pi, N)$, which are selected in a determined way:

- Transition matrix $\boldsymbol{A}$ is an N by N matrix with entries $a_{nn} = 1/N$
- Emission matrix $\boldsymbol{B}$ is an N by M matrix with a Gaussian distribution around its principal diagonal
- Duration matrix $\boldsymbol{D}$ is an N by D_N matrix, where D_N is the maximum number duration steps ($d = 1,2,...,D_N$) in one particular state; the values of $\boldsymbol{D}$ are exponentially decreasing with d
- Initial state vector with length N has the uniform entries $\pi_n = 1/N$
- N is in first place chosen as half of the maximum value of clusters $\boldsymbol{C} = C_t$

The estimates of these parameters, signified by a hat, serve also as inputs. All of the estimated parameters are set to zero before the first iteration.

At first, the Forward step is executed with a first initialization at $t = 1$ and then an induction from $t = 2,3,...,T$. Here, the forward variable $\boldsymbol{\alpha}_t \in \mathbb{R}^{N \times D_N}$ and $P(C_t|\boldsymbol{\lambda}) = r_t$, which assesses the model fitness, are calculated. In the Backward

step, the state S_t and the estimated parameters $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$, and $\hat{\boldsymbol{D}}$ are identified. This step is again divided into an initialization at $t = T$ and induction $t = T-1, T-2, \ldots 1$, though in the reverse chronological order.

After every iteration of the Forward-backward step, the parameters are updated. Thus, first $\hat{\boldsymbol{D}}$ and $\hat{\pi}$ are calculated. After that, all estimated parameters are normalized. The final step in every iteration is the calculation of the overall model likelihood $P(C_1 \ldots C_T|\boldsymbol{\lambda})$. This procedure is iteratively repeated until a desired value for the likelihood $P(C_t|\boldsymbol{\lambda})$ is achieved or a maximum number of iterations is reached.

Table A.2.: Applied Forward-backward step of the HsMM algorithm. Based on the theory introduced in [YK06]

Inputs

$\boldsymbol{A} = a_{nn}$, $\boldsymbol{B} = \boldsymbol{b_n} = b_n(C_t)$, $\boldsymbol{D} = \boldsymbol{d}_n = d_n(d)$, $\boldsymbol{\pi} = \pi_n$, N, $\boldsymbol{C} = C_t$

$\hat{\boldsymbol{A}} = \hat{a}_{nn}$, $\hat{\boldsymbol{B}} = \hat{\boldsymbol{b}}_n = \hat{b}_n(C_t)$, $\hat{\boldsymbol{D}} = \hat{\boldsymbol{d}}_n = \hat{d}_n(d)$, $\hat{\boldsymbol{\pi}} = \hat{\pi}_n$

Forward

Initialization ($t = 1$) and induction ($t = 2,3,...,T$):

$$\boldsymbol{\alpha}_t(n) = \begin{cases} \pi_n \boldsymbol{d}_n & t = 1 \\ s_{t-1}(n)\boldsymbol{d}_n + b^*_{n,t-1}(C_{t-1})\,\boldsymbol{\alpha}_{t-1}(n) & t = 2,3,...,T \end{cases}$$

$$P(C_t|\boldsymbol{\lambda}) = r_t = \sum_{n,d} \boldsymbol{\alpha}_t(n) b_n(C_t)$$

$$\boldsymbol{b}^*_{n,t} = \frac{\boldsymbol{b}_n}{r_t}$$

$$e_t(n) = \alpha_t(n, d = 1)\,\boldsymbol{b}^*_{n,t}$$

$$s_t(n) = \sum_m e_t(n)\, a_{mn}$$

Backward

Initialization ($t = T$) and induction ($t = T-1, T-2, ..., 1$):

$$\hat{a}_{nn} = \begin{cases} \hat{a}_{nn} + e_t(n) & t = T \\ \hat{a}_{nn} + e_t(n)\, e^*_t(n)^T & t = T-1, T-2, ..., 1 \end{cases}$$

$$\gamma_t(n) = \begin{cases} b^*_{n,t}(C_t) \sum_d \boldsymbol{\alpha}_t(n) & t = T \\ \gamma_{t+1}(n) + e_{t+1}(n) s^*_{t+1}(n) - s_{t+1}(n)\, e^*_{t+1}(n) & t = T-1, T-2, ..., 1 \end{cases}$$

$$\hat{b}_n(C_t) = \hat{b}_n(C_t) + \gamma_t(n)$$

$$S_t = \arg\max_n \gamma_t(n)$$

$$\beta_t(n,d) = \begin{cases} b^*_{n,t}(C_t) & t = T \\ s^*_{t+1}(n)\, b^*_{n,t+1}(C_{t+1}) & t = T-1, T-2, ..., 1 \wedge d = 1 \\ \beta_{t+1}(n, d-1)\, b^*_{n,t+1}(C_{t+1}) & t = T-1, T-2, ..., 1 \wedge d > 1 \end{cases}$$

$$e^*_t(n) = \sum_d d_n(d)\, \beta_{t+1}(n,d)$$

$$s^*_t(n) = \sum_m a_{mn}\, e^*_t(n)$$

$$\hat{\boldsymbol{d}}_n = \hat{\boldsymbol{d}}_n + s_{t+1}(n)\, \boldsymbol{\beta}_{t+1}(n)$$

Table A.3.: Final re-estimation step and outputs of the applied HsMM algorithm. Based on the theory introduced in [YK06]

Re-estimation

After every iteration:

$$\hat{\boldsymbol{d}}_n = \hat{\boldsymbol{d}}_n + \pi_n \boldsymbol{\beta}_2(n)$$

$$\hat{\pi}_n = \hat{\pi}_n + \frac{\gamma_1(n)}{\sum_n \gamma_1(n)}$$

$$\pi_n = \frac{\hat{\pi}_n}{\sum_n \hat{\pi}_n}$$

$$\boldsymbol{b}_n = \frac{\hat{\boldsymbol{b}}_n}{\sum_m \hat{\boldsymbol{b}}_n}$$

$$a_{nm} = \frac{\hat{a}_{nm}}{\sum_n \hat{a}_{nm}}$$

$$\boldsymbol{d}_n = \frac{\hat{\boldsymbol{d}}_n}{\sum_d \hat{\boldsymbol{d}}_n}$$

$$P(C_1 \ldots C_T|\boldsymbol{\lambda}) = \prod_{\tau=1}^{T} P(C_\tau|\boldsymbol{\lambda})$$

Outputs

$\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{D}, \boldsymbol{\pi}, S_t, P(C_1 \ldots C_T|\boldsymbol{\lambda})$

A.3 Determining the number of clusters

As presented in Section 3.3.3, the number of clusters, which has to be defined during the training process, is selected with reference to the achieved Silhouette value Sil. Figure A.2 shows an example course of Sil with an increasing number of clusters K. Since the course converges for large values of K to one, a limit ($Sil = 0.9$) is predefined, which indicates a high structure in clustering [Rou87]. The number of clusters K is determined, when the limit is crossed for the first time.

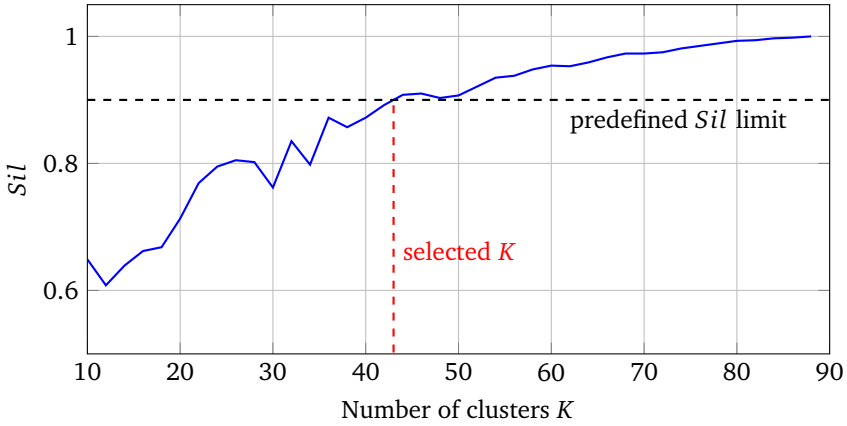


Figure A.2.: Silhouette value evaluation with different numbers of clusters K

A.4 Algorithms for state estimation and model identification

The complete algorithms for both state estimation and model identification are summarized in Table A.4 and Table A.5. For the state estimation, many formulae are adopted from the theory of [YK06] such as the entire Forward step of Table A.2. From the Backward step only the variable $\gamma_t(n) = P(q_t = \boldsymbol{S}_t|\boldsymbol{S}_C)$ is calculated.

Several other differences exist, which arise mainly due to the inspection of the probability distribution captured with the class probability $\boldsymbol{S}_C$ instead of the one-dimensional clusters $\boldsymbol{C}$. Hence, several parameters such as emission matrix $\boldsymbol{B}$ are extended by another dimension. The resulting parameters are symbolized with a tilde. A further difference is that not the entire cluster signal, but only a sliding window of length $\Delta T = 100$ is examined.

The two outputs of this first algorithm are the model likelihood $P(\boldsymbol{S}_C|\boldsymbol{\lambda})$ and the probability vector $\boldsymbol{\gamma}_t$, which captures the most probable damage states. The model likelihood is then transferred to the second algorithm in Table A.5 that calculates the model probability $\boldsymbol{\mu}_k$ in time step k. The formulae are partly taken from the Interacting Multiple Model (IMM) approach, introduced in [Blo84].

After a first normalizing step, where the model likelihood is converted into an estimated model probability $\hat{\boldsymbol{\mu}}$, a reinitializing step begins. Here, the model probabilities of the previous time step are multiplied with a transfer matrix $\boldsymbol{H}$. Similar to the transition matrix $\boldsymbol{A}$ of HsMM, this transfer matrix contains the probabilities of both a change from model m_j to m_i (expressed by the index h_{ji}) and remaining in model m_j (expressed by the index h_{jj}). Thus, by setting h_{jj} high in comparison to h_{ji} (with $j \neq i$), the probability of selecting model m_j again as the most accurate is increased and rapid model changes are suppressed. In this thesis this parameter is defined as $h_{jj} = 0.9$. The weights $\boldsymbol{w}$ are calculated afterwards. Finally, the new model probability $\boldsymbol{\mu}_k$ is determined.

Table A.4.: Applied state estimation algorithm. Based on the theory introduced in [YK06] extended by the classification uncertainty

Inputs

$\lambda = (\boldsymbol{A} = a_{nn}, \tilde{\boldsymbol{B}} = \tilde{\boldsymbol{b}}_{\boldsymbol{n}} = \tilde{b}_n(\boldsymbol{S}_C(t)), \boldsymbol{D} = \boldsymbol{d}_n = d_n(d), \boldsymbol{\pi} = \pi_n, N), \boldsymbol{S}_C = S_C(t,m)$

Forward

Initialization ($t = 1$) **and induction** ($t = T - \Delta T,\ldots,T$)**:**

$$\boldsymbol{\alpha}_t(n) = \begin{cases} \pi_n \boldsymbol{d}_n & t = 1 \\ s_{t-1}(n)\boldsymbol{d}_n + \tilde{b}^*_{n,t-1}(\boldsymbol{S}_{C,t-1})\boldsymbol{\alpha}_{t-1}(n) & t = 2,3,\ldots,T \end{cases}$$

$$P(\boldsymbol{S}_{C,t}|\boldsymbol{\lambda}) = \boldsymbol{r}_t = \sum_{n,d} \boldsymbol{\alpha}_t(n)\tilde{b}_n(\boldsymbol{S}_{C,t})$$

$$\tilde{\boldsymbol{b}}^*_{n,t} = \frac{\tilde{b}_n}{\boldsymbol{r}_t}$$

$$e_t(n) = \alpha_t(n,d=1)\,\tilde{\boldsymbol{b}}^*_{n,t}$$

$$s_t(n) = \sum_m e_t(n)\,a_{mn}$$

Backward

Initialization ($t = T$)

$$\gamma_t(n) = \tilde{b}^*_{n,t}(\boldsymbol{S}_{C,t})\sum_d \boldsymbol{\alpha}_t(n) \quad t = T$$

Outputs

$P(\boldsymbol{S}_{C,T-\Delta T}\ldots\boldsymbol{S}_{C,T}|\boldsymbol{\lambda}),\ \gamma_t(n)$

Table A.5.: Superior algorithm for the calculation of model probability. Inspired by the theory of Interacting Multiple Models (IMM) introduced in [Blo84]

Inputs
$P(\boldsymbol{S}_C \vert m_j)$, $\boldsymbol{H} = h_{ij}$
Normalizing
$\hat{\mu}_{j,k} = \frac{P(\boldsymbol{S}_C \vert m_j)}{\sum_{k=1}^{N_m} P(\boldsymbol{S}_C \vert m_k)}$
Reinitializing
$\hat{\boldsymbol{\mu}}_k^- = \boldsymbol{H}\boldsymbol{\mu}_{k-1}$
$\boldsymbol{w}_k = \frac{h_{ji}\mu_{j,k-1}}{\hat{\mu}_{j,k}^-}$
New model probability
$\boldsymbol{\mu}_k = \boldsymbol{w}_k \hat{\boldsymbol{\mu}}_k$
Output
$\boldsymbol{\mu}_k$

A.5 Parameters of the proposed algorithm

The following table presents all parameters of the proposed algorithm. It is divided into input parameters, which have to be preselected, the parameters, which are the result of the optimization steps, filter parameters (mainly during feature generation), and other settings.

Table A.6.: Parameters of the proposed algorithm

Input parameters	
Number of training data N_{td}	4, 6, 8, 10
Number of features	3, 4, 5, 5 (corresponds to N_{td})
Net probability	likelihood-based, 0 %-100 %
Self-learning parameters	
Number of damage states N	Optimized by BIC criterion
Number of damage classes M	Optimized by Sil criterion
Filter settings	
Notch filter - cutoff frequency (time signal)	$n \cdot (48\,\text{Hz} - 52\,\text{Hz})$
Frequency bands	5 Hz
Low-pass filter - cutoff frequency (time signal)	1875 Hz
Moving average filter (continuous features)	12 steps
Moving average filter (discrete features)	100 steps
Interacting multiple model self transition probability	0.9
Other settings	
Discretizing levels	5
Mutual information - cutoff	0.9
Sliding window ΔT - diagnosis	400 steps
Number of samples n_s	1000

A.6 Additional plots for verification

This section provides additional plots that support the findings during the verification in Section 3.6. In the first plot in Figure A.3, the derived BIC course during one training process is plotted. By finding the minimum BIC value, the number of damage states is determined.

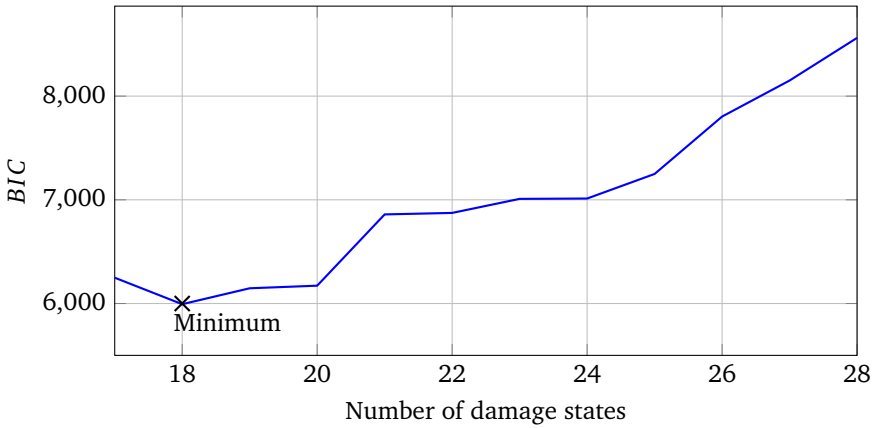


Figure A.3.: Optimizing the number of damage states N for HsMM by evaluating the BIC values

In the second plot, one example of the feature subset, identified during the feature selection step, is plotted in Figure A.4. Four features are depicted for two different runs. A similarity between both runs can only be recognized in feature 1.

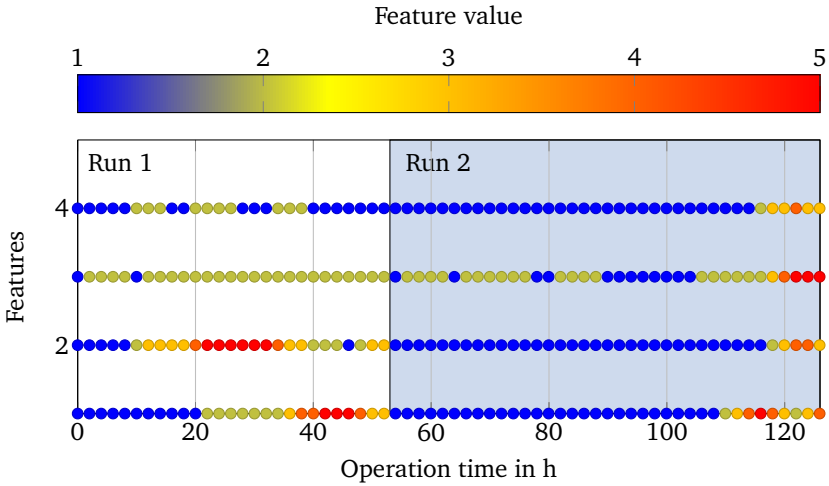
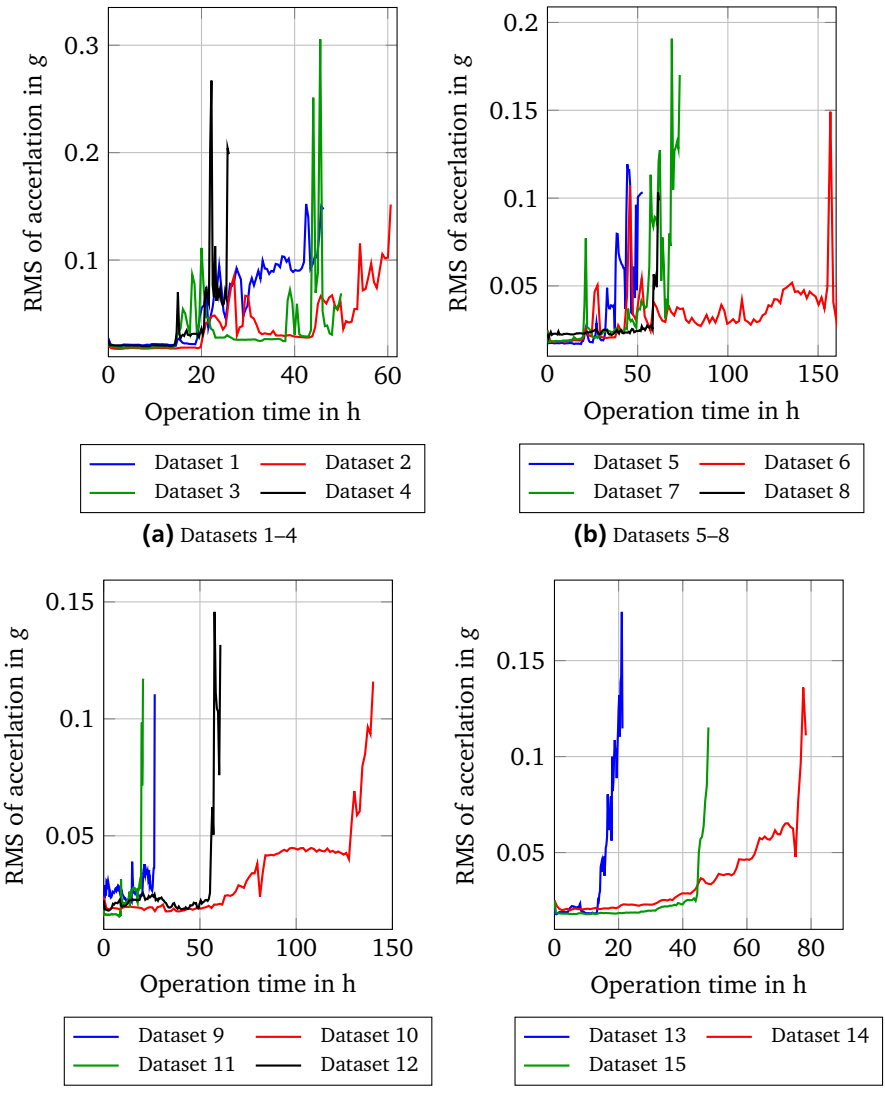


Figure A.4.: Feature subset Φ of two runs

A.7 All datasets

In the following figure, all applied datasets for validation are depicted by the root mean square (RMS) value of the acceleration signal. The 15 datasets are sorted in chronological order of generation.



(a) Datasets 1–4

(b) Datasets 5–8

(c) Datasets 9–12

(d) Datasets 13–15

Figure A.5.: Acceleration signals (RMS) of all applied datasets (in chronological order)

A.8 Descriptive damage analysis

In Table A.7 and Table A.8, the changes of the single bearing components due to the run to failure trials are recorded. Several characteristics like the condition of the lubrication or the appearance of corrosion are summarized for each dataset to distinguish between different damage processes. Due to corrosion, the shaft was exchanged (see Section 5.1.3) after the sixth trial so that in Table A.7 there is a further distinction of *old* and *new* shaft. Besides the abort criterion, the appearance of characteristic, brown *marks* mainly on the outer raceways of the bearings is registered in Table A.8. The origin of these locally contained marks remains unknown, but it is assumed to be a sort of corrosion.

Table A.7.: Damage description of the applied datasets after trial (Part I)

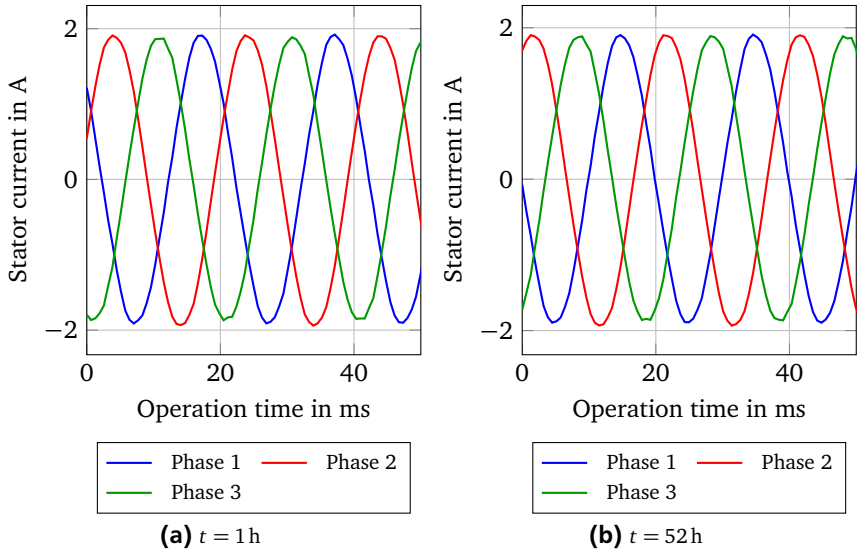
Dataset	Shaft	Color change				Chatter marks		Roughening			Lubrication
		Cage	Inner	Outer	Balls	Inner	Outer	Inner	Outer	Balls	Burnt out
1	old	x			x	x		x	x		x
2	old	x			x	x	x	x	x		x
3	old	x		x	x		x		x	x	x
4	old			x	x						x
5	old	x			x		x	x	x		x
6	old	x		x	x			x	x	x	x
7	new	x			x		x	x	x	x	x
8	new		x	x	x				x	x	
9	new		x	x	x				x	x	
10	new		x	x	x			x	x	x	x
11	new		x	x							x
12	new	x	x	x	x			x	x	x	x
13	new							x	x		
14	new	x	x	x	x			x	x		x
15	new	x	x	x	x			x	x		x

Table A.8.: Damage description of the applied datasets after trial (Part II)

Dataset	Particles			Corrosion			Cage	Marks		Abort criterion	
	Inner	Outer	Balls	Inner	Outer	Balls	Broken	Inner	Outer	Acceleration	Current
1						x			x	x	
2	x	x		x						x	
3	x	x	x	x		x		x	x	x	
4	x	x	x					x	x	x	
5	x	x	x	x				x		x	
6	x	x	x	x			x	x	x	x	
7	x	x	x							x	
8	x	x						x	x		x
9	x	x	x		x						x
10	x	x	x		x		x		x		x
11	x	x	x				x				x
12		x			x				x		x
13	x	x	x								x
14	x	x	x						x		x
15	x	x	x		x		x		x		x

A.9 Derived time signals of motor current

In Figure A.6, an extract of all three stator current phases in time domain is plotted for an early point in the life cycle of the component (see Figure A.6a) and for a point near the component's failure (see Figure A.6b). The examination of both plots indicates no changes in the magnitude of the phases. Thus, a transformation as suggested in Section 3.2.1 into the frequency domain is necessary to assess the actual health status of the component.



(a) $t = 1\,\mathrm{h}$ **(b)** $t = 52\,\mathrm{h}$

Figure A.6.: Stator current signals of all phases at two different points in the component's life cycle (dataset 5)

A.10 Additional plots for validation

Similar to Section A.6, additional plots for both validations of Chapter 5 are presented in this section. Besides the diagnosis step in Section A.10.1, the focus of the other subsections is on the prognosis validation.

A.10.1 Diagnosis

For a better insight into the training process and the related characteristics of Section 5.2, the results of the training process for four training runs are illustrated in Figure A.7.

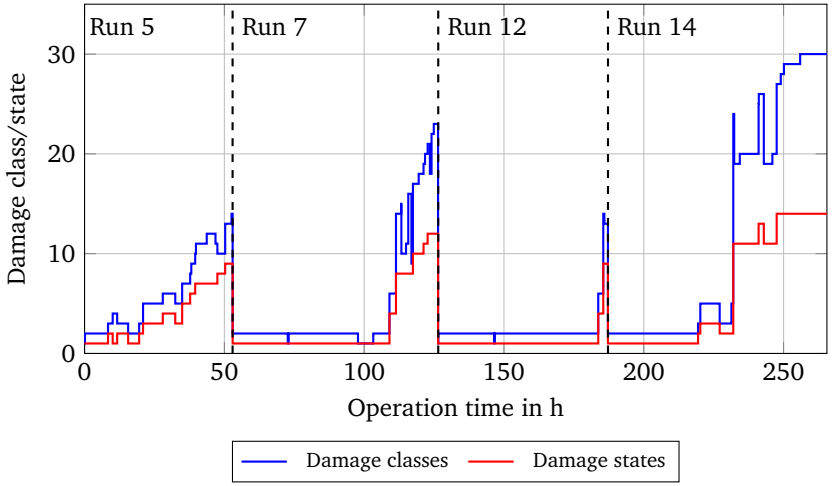
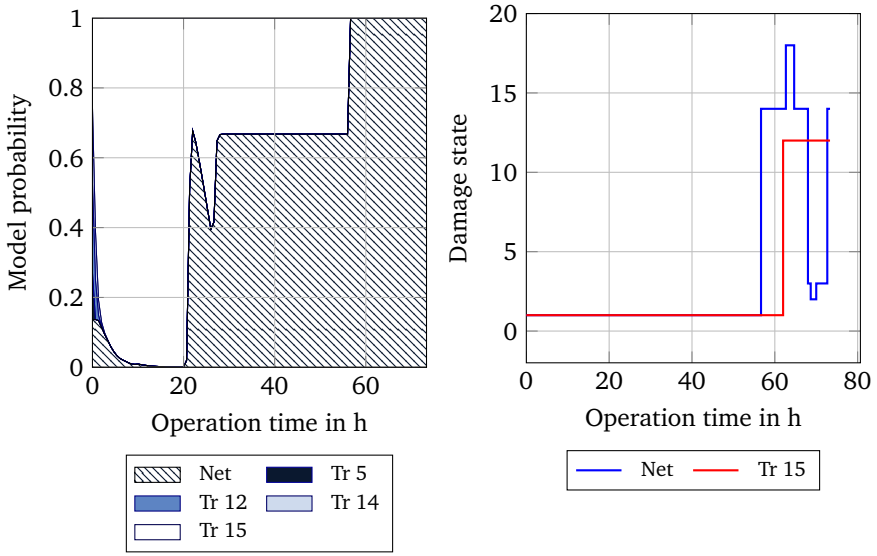


Figure A.7.: Training of HsMMs with four different runs in context of validation

A.10.2 Load-based prediction approach

For a closer look into the two basic steps of fault diagnosis, the model identification and state estimation during validation is plotted in Figure A.8.



(a) Model identification

(b) State estimation

Figure A.8.: Model identification and state estimation during validation with tested dataset 7

A.10.3 Different parameters

Two further parameters are examined in this subsection to support the findings in Section 5.3. First, the model probability of the net during the leave one out cross validation with the applied subset is plotted in Figure A.9. The second plot in Figure A.10 indicates exemplarily, which features are selected during an increasing feature dimension.

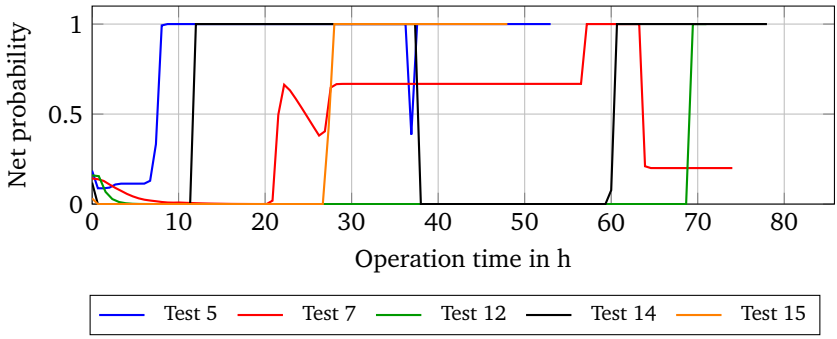


Figure A.9.: Likelihood-based net probability in context of leave one out cross validation with datasets 5, 7, 12, 14, and 15

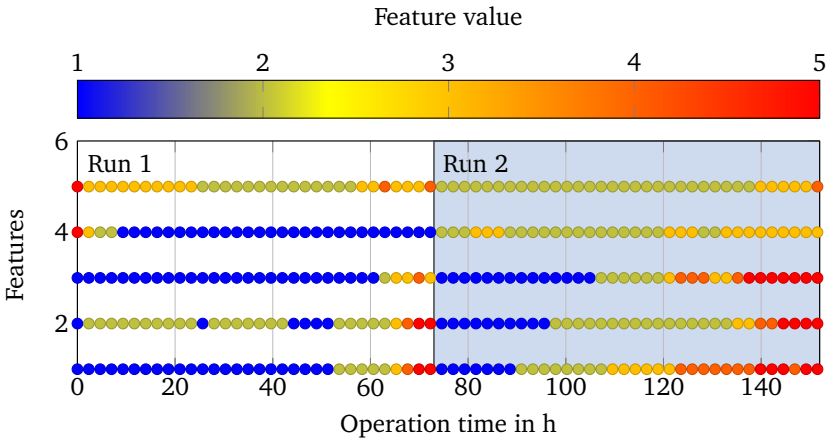
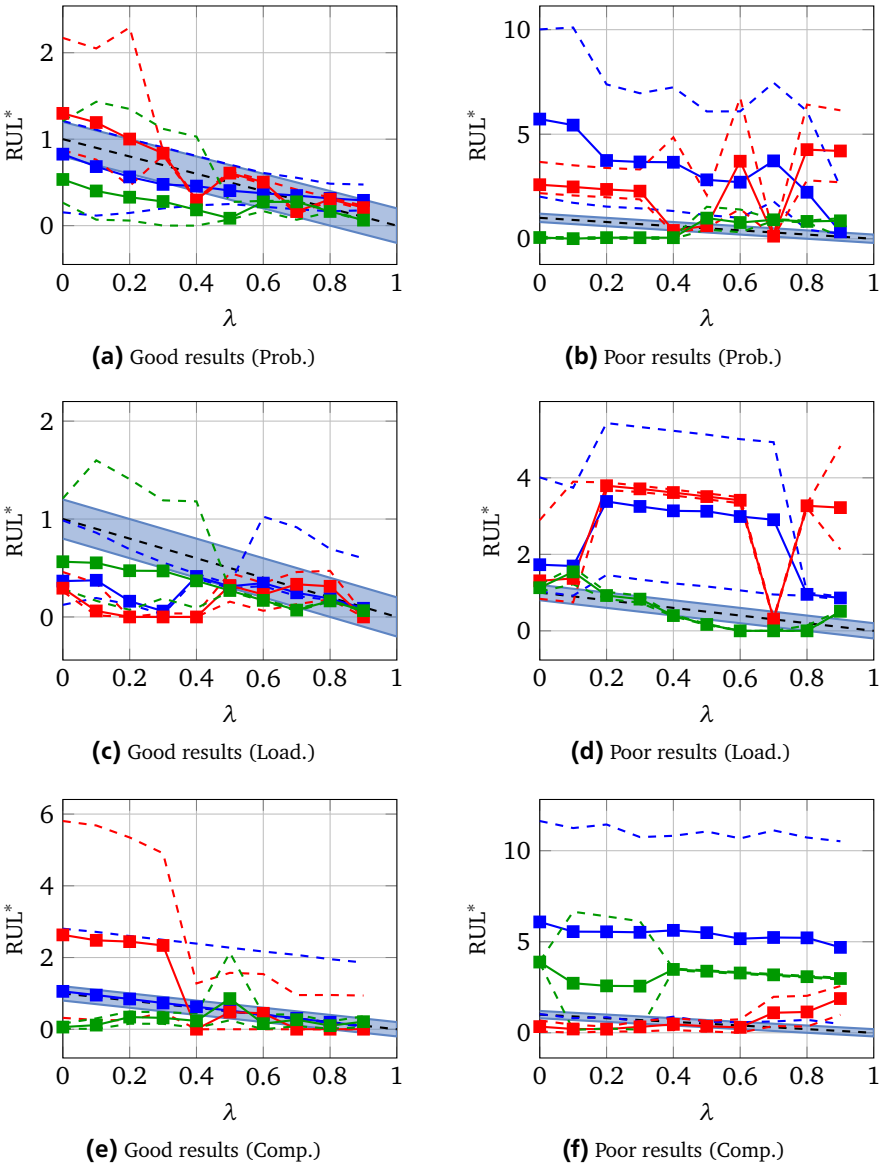


Figure A.10.: Feature subset Φ of two runs during the variation of feature dimension of Section 5.3.5

A.10.4 Validation with all datasets

Figure A.11 gives an insight into the achieved prognostic performances of the three applied HsMM approaches during the validation with all datasets. The plots are divided into good results (left column), where the actual RUL is estimated accurately or the particular algorithm shows a good convergence. In contrary, the right side exhibits test cases, when the algorithms have issues to identify the true RUL.



(a) Good results (Prob.)

(b) Poor results (Prob.)

(c) Good results (Load.)

(d) Poor results (Load.)

(e) Good results (Comp.)

(f) Poor results (Comp.)

Figure A.11.: Selected good and poor prediction results during the validation with all available datasets

A.10.5 Metrics for number of training datasets

In Table A.9, the achieved performance metrics for an increasing number of training data for the three applied HsMM approaches are summarized. These values form the base for the mean calculation of Section 5.4.2.

Table A.9.: Averaged performance metrics with respect to the varying number of training data N_{td}

	$\overline{\text{PH}}$				$\overline{\text{MAPE}}$				$\overline{\text{MPAB}}$				$\overline{\text{SSD}}$			
No. Tr. Data →	4	6	8	10	4	6	8	10	4	6	8	10	4	6	8	10
Prob. HsMM	0.11	0.10	0.08	0.10	173	175	166	179	24	26	25	22	276	277	300	361
Load. HsMM	0.09	0.14	0.08	0.08	152	158	175	180	26	27	26	24	270	266	297	359
Comp. HsMM	0.06	0.04	0.05	0.07	385	261	250	189	27	26	22	28	255	301	290	288