LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
LUT School of Energy Systems
Energy Technology

Juha Juselius

ADVANCED CONDITION MONITORING METHODS IN THERMAL POWER PLANTS

Master's Thesis

Supervisor: D.Sc.(Tech) Jouni Ritvanen

ABSTRACT

Lappeenranta University of Technology
LUT School of Energy Systems
Energy Technology

Juha Juselius

Advanced Condition Monitoring Methods in Thermal Power Plants

Master's thesis

2018

77 pages, 27 figures and 9 tables

Examiners: D.Sc.(Tech) Jouni Ritvanen
Professor, D.Sc.(Tech) Timo Hyppänen
Instructor: M.Sc. (Eng.) Ilkka Salmensaari

Keywords: condition monitoring, power plant maintenance, artificial neural networks

Power plant maintenance has been traditionally organized according to corrective and preventive maintenance strategies, which both have their limitations that can lead to high costs due to low availability and inefficiently timed maintenance actions. Condition based maintenance is considered an advanced maintenance strategy and is already applied in many critical components in modern power plants. Extending condition monitoring and more efficient maintenance practices to a wider range of equipment is desirable but requires advanced condition monitoring methods that can detect changes in process parameters indicating faults and malfunctions. These methods include different mathematical models that can be classified as physics-based, data-driven and hybrid models.

This thesis presents the theoretical background of different modelling approaches in condition monitoring applications. The theoretical part of this work also presents different maintenance strategies and the role of condition monitoring in power plant maintenance. The main objective of this thesis is to evaluate a commercial condition monitoring software based on recursive neural networks, called Intelligent Health Monitor, in a case study where boiler's increased flue gas temperature is monitored. 96 models with different training parameters are created and evaluated during training, healthy state monitoring and fault detection phases. The results showed that reliable models with good fault detection capability can be trained with automatically chosen input variables based on principal component analysis and correlation analysis. The results also showed great variation and underlined the importance of model verification. Based on the case study results and literature, several aspects that need to be considered when applying artificial neural network based condition monitoring software are discussed.

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
LUT School of Energy Systems
Energiatekniikan koulutusohjelma

Juha Juselius

Kehittyneet kunnonvalvontamenetelmät lämpövoimalaitoksissa

Diplomityö

2018

77 sivua, 27 kuvaa ja 9 taulukkoa

Tarkastajat: TkT Jouni Ritvanen
Professori Timo Hyppänen

Ohjaaja: DI Ilkka Salmensaari

Hakusanat: kunnonvalvonta, voimalaitoksen kunnossapito, neuroverkot

Voimalaitosten kunnossapito on perinteisesti noudattanut korjaavia tai ennakoivia kunnossapitostrategioita. Perinteisten strategioiden puutteiden ja niistä aiheutuvien korkeiden kustannusten vuoksi kuntoon ja kunnonvalvontaan perustuvia kunnossapitostrategioita tarvitaan. Monien kriittiseksi luokiteltujen komponenttien kunnossapito perustuu jo kunnonvalvonnan tuloksena saatuun komponentin kunnon arvioon, mutta kunnonvalvonnan kattavuutta tulisi laajentaa myös ei-kriittiseksi luokiteltuihin komponentteihin. Tämä edellyttää kehittyneitä kunnonvalvontamenetelmiä, kuten matemaattisia mallinnusmenetelmiä, joilla pystytään havaitsemaan prosessidatasta poikkeavuuksia, jotka indikoivat vikatiloja. Kunnonvalvonnassa käytetyt mallit voidaan luokitella fysikaalisiin malleihin, data-pohjaisiin malleihin ja hybridimalleihin.

Tämän työn teoriaosassa esitellään eri kunnossapitostrategiat, kunnonvalvonnan merkitys voimalaitosten kunnossapidossa sekä eri mallinnusmenetelmien teoreettinen tausta. Työn päätavoitteena on testata ja arvioida kaupallista neuroverkkoihin perustuvaa kunnonvalvontaohjelmistoa Intelligent Health Monitoria. Ohjelmiston soveltuvuutta arvioidaan case-tapauksessa, jossa voimalaitoskattilan savukaasun lämpötila on noussut laskien kattilahyötysuhdetta. Savukaasun lämpötilaa seurataan 96 mallilla, joiden tarkkuus arvioidaan kolmena ajanjaksona: koulutusjakso, vikaantumattoman tilan jakso sekä vikatilanteen jakso. Tulokset osoittavat, että luotettava neuroverkkopohjainen malli voidaan luoda käyttämällä automaattisesti valittuja itsenäisiä muuttujia. Tulokset kuitenkin vaihtelivat huomattavasti mallien välillä, mikä korostaa käytettävien mallien verifioinnin tarpeellisuuden. Case-tapauksen tulosten sekä kirjallisuuden perusteella nostetaan esiin useita näkökulmia, jotka tulee huomioida käytettäessä neuroverkkopohjaista ohjelmistoa kunnonvalvonnassa.

FOREWORD

This thesis was done for Fortum Power Solutions as an independent study as part of Fortum's contribution to ECSEL Joint Undertaking MANTIS. Firstly, I would like to thank Fortum Power Solutions for providing me this great opportunity to work on this interesting topic and exciting work environment. I want to express my gratitude towards all the Fortum's experts who have helped and guided me throughout this thesis. Especially I want to thank Ilkka Salmensaari, Matti Kaija and Matti Visuri for all their valuable guidance and feedback. I also want to thank my supervisor Jouni Ritvanen for all the provided instructions and comments.

Last, but certainly not least, I wish to thank my parents and family for their unconditional support from the beginning of my studies. Naturally, these special thanks are extended to my beloved graphic designer.

Espoo, 31st of March 2018

Juha Juselius

TABLE OF CONTENTS

NOMENCLATURE

$\mathbf{E}$	residual matrix	[-]
$\mathbf{F}$	residual matrix	[-]
f	measured value	[-]
$\mathbf{I}_m$	identity matrix	[-]
J	performance index	[-]
k	discrete time	[-]
$\mathbf{L}$	diagonal matrix	[-]
L	number of hidden layer	[-]
l	number of retained principal components	[-]
m	number of process variables	[-]
N_h	number of hidden neurons	[-]
N	input-target relation	[-]
n	number of observations	[-]
n_i	number of inputs	[-]
n_o	number of outputs	[-]
n_p	number of input samples	[-]
o	output of a single neuron	[-]
$\mathbf{P}$	matrix of orthogonal vectors	[-]
$\widehat{\mathbf{P}}$	matrix of retained eigenvectors	[-]
$\widetilde{\mathbf{P}}$	matrix of ignored eigenvectors	[-]
$\boldsymbol{p}$	input loading vector	[-]
Q	heat energy	[W]
$\boldsymbol{q}$	output loading vector	[-]
$\mathbf{S}$	input score matrix	[-]
SS	sum of squares	[-]
$\boldsymbol{s}$	input score vector	[-]
T	temperature	[°C]
T_{f}	time to failure	[s]

$\mathbf{T}$	matrix of transformed variables	[-]
$\hat{\mathbf{T}}$	matrix of retained principal components	[-]
$\tilde{\mathbf{T}}$	matrix of ignored principal components	[-]
t	equipment current age	[s]
$\mathbf{U}$	output score matrix	[-]
$\boldsymbol{u}$	input vector	[-]
$\mathbf{W}$	weight coefficient matrix	[-]
$\boldsymbol{w}$	weight coefficient vector	[-]
$\mathbf{X}$	input data matrix	[-]
$\hat{\mathbf{X}}$	matrix of modeled variation	[-]
$\mathbf{Y}$	output matrix	[-]
$Y(t)$	past condition profile up to current time	[-]
$\boldsymbol{y}$	output vector	[-]
y	predicted value	[-]
z	target for single neuron	[-]

Subscripts

abs	absorbed
i	input
reg	regression
res	residual
tot	total

Greek Letters

α	learning rate	[-]
η	efficiency	[-]
σ	vector valued activation function	[-]

Abbreviations

AI	artificial intelligence
ANN	artificial neural network
Auto	automatic

CBM	condition based maintenance
CI	computational intelligence
CM	condition monitoring
CMS	condition monitoring system
ECSEL	Electronic Components and Systems for European Leadership
FDI	fault detection and isolation
FLS	fuzzy logic system
GRNN	generalized regression neural network
IHM	Intelligent Health Monitor
IVS	input variable selection
MAE	mean absolute error
Man	manual
MSE	mean squared error
MIV	mean impact value
MLP	multilayer perceptron
O&M	operation and maintenance
OSA-CBM	Open Standard Architecture Condition Based Maintenance
PCA	principal component analysis
PLS	partial least squares
PoF	physics-of-failure
PPM	predetermined preventive maintenance
RBF	radial basis function
RMSE	root mean squared error
RUL	remaining useful life
SOM	self-organizing map

1 INTRODUCTION

The majority of the world's energy consumption is covered with conventional energy production methods, such as hydro power turbines, nuclear fission, and combustion of gas, oil or coal, see Figure 1. In power production, the actual production process is often optimized, i.e. the efficiency is relatively high and not easy to increase without high investment costs. At the same time, maintenance and availability are identified as crucial part of successful operation. Low availability can lead to major costs in the form of lost production hours and expensive repair work and therefore, the most important goal of a power operator is to maximize plant's availability and performance with minimum costs (Kaija 2016, 1). However, the production process and maintenance cannot be separated. Running the energy production process inefficiently can cause significant costs if the production equipment would suffer from wear and faults as the result of misuse. Consequently, proper information regarding the production process and condition of the production equipment is identified as a vital part of process development.

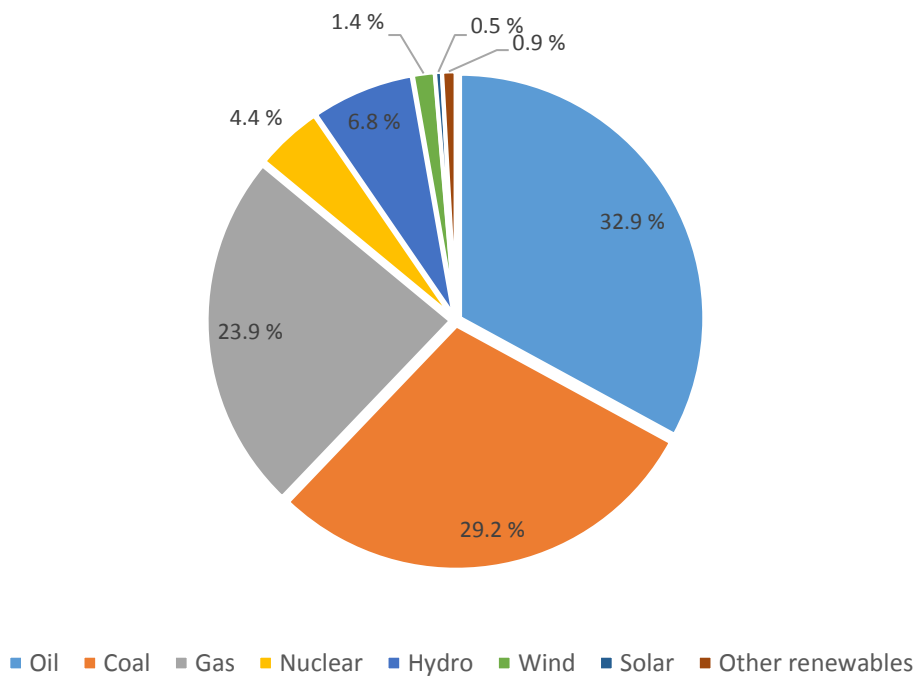


Figure 1. The world's comparative primary energy consumption in 2015 (The World Energy Council 2016, 4).

Traditionally, industrial maintenance has been organized according to two strategies, namely corrective and preventive maintenance. In corrective maintenance, actions are taken to restore equipment back to operational state after failures, whereas preventive maintenance aims to prevent faults and malfunctions with planned maintenance actions before faults occur. Preventive maintenance actions, such as repairs and overhauls, are scheduled in advance usually based on time or hours run. A fixed schedule for maintenance actions is often inefficient, as maintenance actions can be carried out without actual need, or too late when equipment performance has already degraded. To minimize the limitations and resulting high costs of traditional maintenance strategies, see Figure 2, maintenance strategies based on equipment's condition are required. As a result, the focus of maintenance strategies has shifted from corrective and scheduled preventive maintenance to condition based maintenance (CBM). This change has come from both technological developments and from maintenance ideology transition. Industrial maintenance ideology has evolved from non-issue "necessary evil" to strategic concern creating value to different stakeholders. Power sector makes no exception and recognizes CBM as the most attractive maintenance approach. (Kobbacy & Murthy, 2008, 21; Martha de Souza 2012, 129)

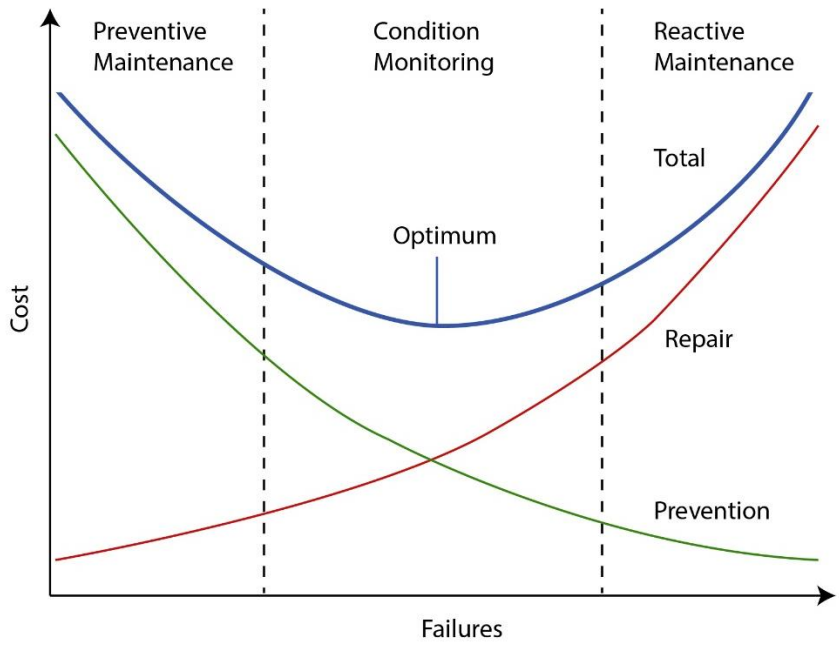


Figure 2. Schematic illustration of maintenance cost distribution between maintenance strategies. Adopted from Akula et al. (2017, 59).

Continuous operation, lost production during outages and very expensive capital equipment leading to limited redundancy are characteristic to power plants. In general, commercial earning of a power generator is dependent on the grid availability and operating efficiency, and therefore, availability is often more valued criterion than reliability. (Chanda & Mukhopaddhyay, 2016, 4). Power plants usually follow a combination of corrective and preventive maintenance strategies augmented with predictive maintenance actions, meaning that the equipment condition is predicted in a future state, on some critical components (Mobley 2002, 2–4). With the rapid development of sensor and data acquisition technologies, signal processing and computational tools, it is desirable to extend condition monitoring to cover a wider range of process equipment to enhance power plant's availability and feasibility.

CBM is premised on condition monitoring activities to determine the physical state of a component of a system. The condition assessment is used to schedule preventive maintenance actions or to change operation mode to decrease unwanted degradation and wear of equipment. The Open Standard Architecture Condition Based Maintenance (OSA-CBM) is a widely used framework for industrial CBM. The framework, illustrated in Figure 3, is designed by MIMOSA, which is an organization involved in CBM standard development (Shin & Jun, 2015, 123).

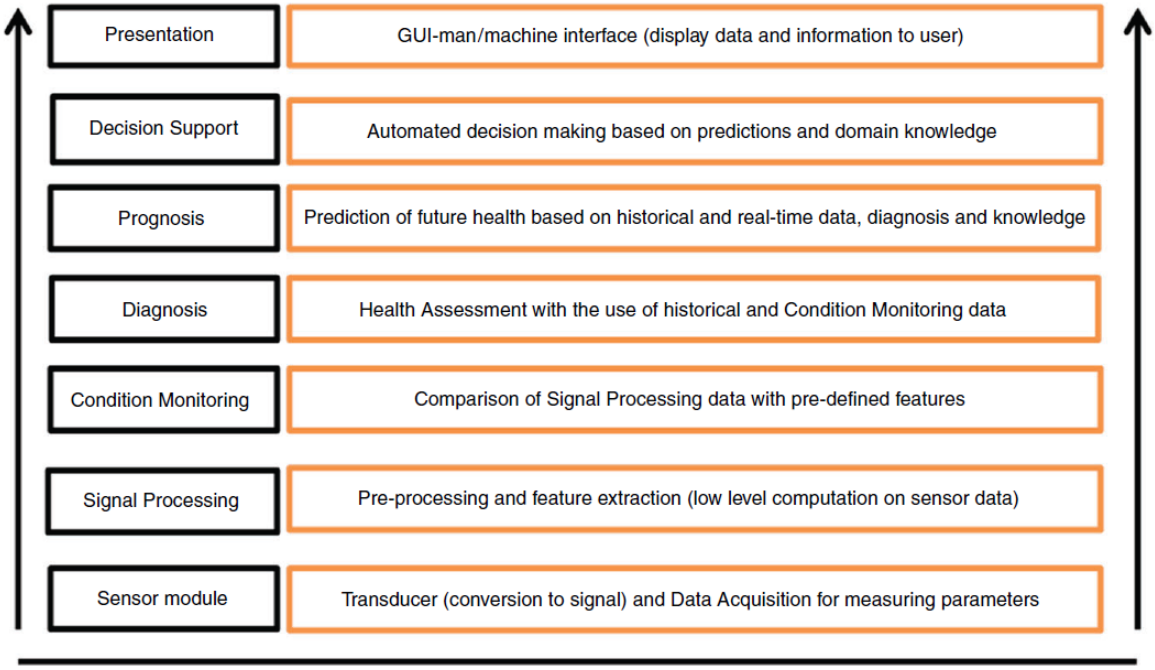


Figure 3. The OSA-CBM framework (Thurston & Lebold, 2001).

The framework includes seven layers describing the whole CBM process from data acquisition to presenting the results. The first layer, i.e. sensor module consists of data acquisition, which includes measurement devices and transducers for signal conversion. The second layer, signal processing, is referred as data preprocessing in this thesis. This layer includes several techniques, such as filtering and sampling, to enhance the data's reliability and to transform the raw data into a more convenient format for further processing. Preprocessing can be integrated to data acquisition with intelligent sensors or it can be treated separately with additional modules. Condition monitoring layer gathers the preprocessed data to be compared with predefined features. Diagnostic layer uses condition monitoring data to obtain a health assessment of the monitored item. Prognostic layer predicts the item's health in a future state based on the results of condition monitoring and diagnostic layers. Execution of maintenance actions is decided on decision support layer, which also considers available recourses, such as spares, logistics and manning. The final layer, namely presentation, includes the man/machine interface used to display data and information to the user. This layer may query all other layers. (Niu et al. 2010, 788; Shin & Jun, 2015, 123, Bousdekis et al. 2015, 1230)

Traditionally, condition monitoring is applied by setting fixed threshold values to monitored parameters, and deviations from the set thresholds triggers an alarm. However, selecting appropriate alarm limits is often challenging, and relies on human experience and knowledge. Advanced condition monitoring methods, as understood in this thesis, are computational models that detect abnormalities automatically without human-effort and indicate the health of the monitored equipment with minimal human-aid. These methods are used in the condition monitoring layer, diagnostic layer and prognostic layer.

1.1 Research goal

The research goal of this thesis is to study advanced condition monitoring methods that include computational models used as part of CBM. The aim is to study methods and tools that are not limited to certain critical equipment, but instead allow plant-wide monitoring. The focus is on computational models used to detect abnormalities, while other modules of

condition monitoring systems (CMS), such as sensor technology and data transfer, are outside the scope of this thesis. The following hierarchical terminology is adopted and used in this thesis:

> **Approach** is a combination of principles and methods defining a broad direction in solving engineering problems. Examples of approaches in the context of this thesis are physics-based modeling and data-driven modeling.
>
> **Methods** or their combinations compose the chosen approach. A method is a more targeted way than approach of describing how the problem is solved. Examples of methods in the context of this thesis are statistical models and artificial neural networks.
>
> **Techniques** and **tools** form the solution method on a concrete level describing the precise strategy used. Examples of techniques and tools in the context of this thesis are principal component analysis in the case of statistical methods, and multilayer perceptrons describing the neural network architecture.

The main objective of this thesis is to evaluate a neural network based commercial condition monitoring software Intelligent Health Monitor by Algorithmica technologies Gmbh. (Algorithmica technologies GmbH, 2018). The software is evaluated in a case study on a combined cycle gas turbine power plant. This thesis presents one of the case studies, in which a boiler's gradually increased flue gas temperature is monitored. Based on the case study results and literature, several aspects that need to be considered when applying artificial neural network based condition monitoring software are discussed answering the research question:

> *How to utilize artificial neural network based condition monitoring software to support thermal power plant operation and maintenance?*

Several advanced condition monitoring methods are applied to most critical power plant components. Typically, CBM is applied to rotating machinery, such as steam and gas tur-

bines, pumps and compressors and electric motors and generators. Other common applications are bearings of different machinery, piping systems, heat exchangers and conveyor systems (Kaija 2016, 13–19). Research related to condition monitoring methods has mostly focused on fault diagnosis and prognosis, which both require clear differentiation between different faults. Therefore, the research has also mainly focused on specific components with well-known fault mechanisms. Numerous such applications have been well reviewed in the literature. For instance, Henao et al. (2014) review diagnostics techniques of rotating electrical machines, Leite et al. (2018) review prognostic methods of wind turbine maintenance, Tahan et al. (2017) review diagnostics and prognostics methods of gas turbines and El-Thalji & Jantunen (2015) review prognostics methods of rolling element bearings.

1.2 Research structure

The theoretical framework of this thesis consists of three sections. Section 2 presents the common maintenance strategies used in thermal power plants. In Section 3, condition monitoring as a crucial part of advanced maintenance is presented. Stages and objectives of condition monitoring and commonly used condition monitoring measurement techniques are introduced. Because of the enormous breath of maintenance as a research field, the objective of these sections is not to provide in depth analysis of maintenance or condition monitoring, but merely to introduce these topics. Section 4 gives an overview of some standard data preprocessing techniques. In Section 5, three different modeling methods used in condition monitoring are presented and discussed. Physical process models, data-driven models and hybrid models are addressed on theoretical level. Section 6 presents a case study where increased flue gas temperature is monitored using commercial condition monitoring software Intelligent Health Monitor (Algorithmica technologies GmbH, 2018). In Section 7, practical aspects of using artificial neural network based condition monitoring as part of a condition monitoring system are discussed. Finally, conclusions are presented, and future work proposed in Section 8.

1.3 Fortum Power Solutions

Fortum is a Finnish energy company with over 8000 (2016) personnel worldwide and a revenue of 3600 M€ (2016). The Power Solutions unit offers operation and maintenance (O&M) and related expert services and solutions to owners and developers of power and heat assets

while employing some 300 personnel (2016). The customer base of the Power Solutions unit consists of both Fortum owned or co-owned plants and completely independent power plant owners around the world. The main service areas are O&M services, burners and emission control systems, power plant performance improvements, turbine and generator services and IT-systems. (Fortum 2017a; Fortum 2017b)

The Power Solutions O&M unit operates and maintains power plants that are not owned by Fortum. The O&M services utilize the O&M principles developed for Fortum owned and serviced plants to offer O&M scheme optimization and audits as a service, including criticality analysis, personnel training and strategy reviews. Most of the O&M operations provided by Fortum are based on the TOPgen concept, or its adaptation. The concept includes setting up an efficient O&M organization, management systems and selected IT tools to maintain the plant availability and performance at an optimal level during the whole life cycle of the plant. (Fortum 2017b)

The performance service unit provides services to improve technical and economic performance of power plants. The services include process analysis and modeling, plant operation monitoring and optimization work delivery. The Turbines and Generators unit offers maintenance and condition management services for power plant turbines in the form of maintenance, inspections, condition management and consultation. The IT Systems unit provides multiple tools to integrate in the power industry services. The four main products are TOPi process monitoring, Solvo process simulation, Apros dynamic process simulation and Maximo asset management software. (Ibid)

This thesis is done for Fortum Power Solutions as an independent study as part of Fortum's contribution to ECSEL Joint Undertaking MANTIS. 47 European industrial and academic partners are contributing to the MANTIS program whose primary objective is to develop a Cyber Physical System based Pro-active Maintenance Service Platform Architecture enabling Collaborative Maintenance Ecosystems (MANTIS 2015).

2 POWER PLANT MAINTENANCE

European Standard EN 13306 (2001) defines maintenance as the combination of all technical, administrative and managerial actions during the life cycle of an item intended to retain it in, or restore it to, a state in which it can perform the required function. The role and importance of maintenance can be extended, and maintenance can be seen contributing not only to profitable business, but to the aim of sustainable development in society, including environmental, energy saving, safety, and economical aspects (Holmberg et al. 2010, 5). Regardless of the industry, there is a consensus that maintenance is much more than a purely technical matter and that a proper maintenance program is vital to attain business, environmental and safety requirements. (Kobbacy & Murthy, 2008, 26; Al-Turki et al. 2014, 5)

Maintenance consists of a vast set of problems and operational implications, which makes the various approaches to maintenance diverse and difficult to schematize in a general manner. Traditionally, maintenance is classified into corrective and preventive maintenance. Besides types of maintenance actions, also multiple strategic and philosophical approaches to maintenance, such as e.g. total productive maintenance that focuses on constant development of the overall equipment effectiveness (Wireman 2004, 14) and reliability centered maintenance that focuses on determining the most effective maintenance approach for individual equipment (Moubray 1997, 7–8), have been developed and implemented in different industries. (Holmberg et al. 2010, 1; Fedele 2011, 33)

2.1 Maintenance terminology

Fault and failure are two fundamental terms describing the state of a system or a component. Fault is considered as an abnormal change of system behavior, which deteriorates the performance of the system. Faults can be classified as illustrated in Figure 4. Abrupt faults occur rapidly, when the parameter indicating faulty behavior abruptly rises to a new constant level. Incipient faults develop gradually over time, for example, in the case of slow degradation of a component. Intermittent faults have a repeated pattern, i.e. they occur and disappear repeatedly. A failure is a consequence of progressing incipient fault over time demolishing the normal operation of the system (Sobhani-Tehrani & Khorasani, 2009, 1–2). (Patan 2008, 9)

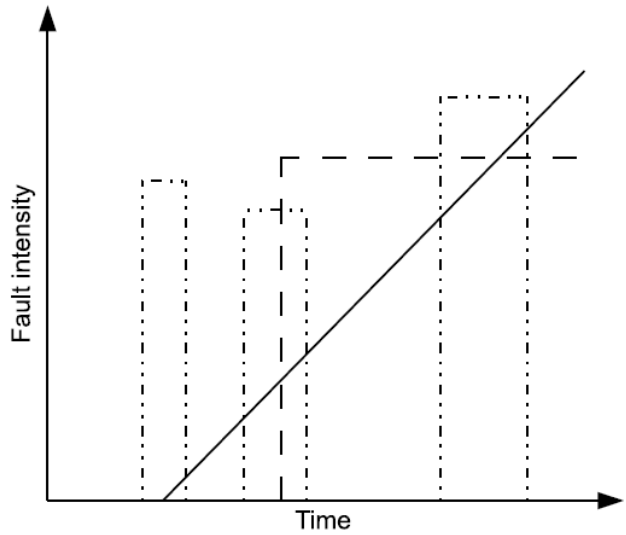


Figure 4. Fault classification: abrupt (dashed), incipient (solid) and intermittent (dash-dot) (Patan 2008, 9).

Maintenance is still a relatively young branch of science, and while some terms have unambiguous definitions, there is still a lot of confusion with maintenance terminology both in literature and in practice. Many terms are used differently between sources and misapprehensions are not uncommon. (Kobbacy & Murthy, 2008) To avoid any confusion, the following elementary maintenance terms are presented and adopted in this thesis.

Maintenance policy indicates the overall attitude assumed in relation to maintenance problems. Examples of maintenance policies are total productive maintenance and reliability centered maintenance. Maintenance policy can be clarified in the use of various strategies.

Maintenance strategy follows maintenance policy in hierarchy. Maintenance strategy is an operational approach to maintenance problems based on the criteria provided by the adopted maintenance policy. Maintenance strategy determines the triggering mechanism for maintenance actions. Examples of maintenance strategies are preventive and predictive maintenance strategies.

Maintenance actions are basic maintenance interventions and tasks performed by a technician. Examples of maintenance actions are inspections, repairs and overhauls.

As stated above, literature still lacks unambiguous terminology and a lot of confusion exists. For instance, Kobbacy & Murthy (2008), Fedele (2011) and Al-Turki et al. (2014) all give different definitions to maintenance terms.

2.2 Maintenance strategies

According to EN 13306 (2001), maintenance can be classified into corrective maintenance and preventive maintenance. Both maintenance types can be classified further into sub-categories, as presented in Figure 5.

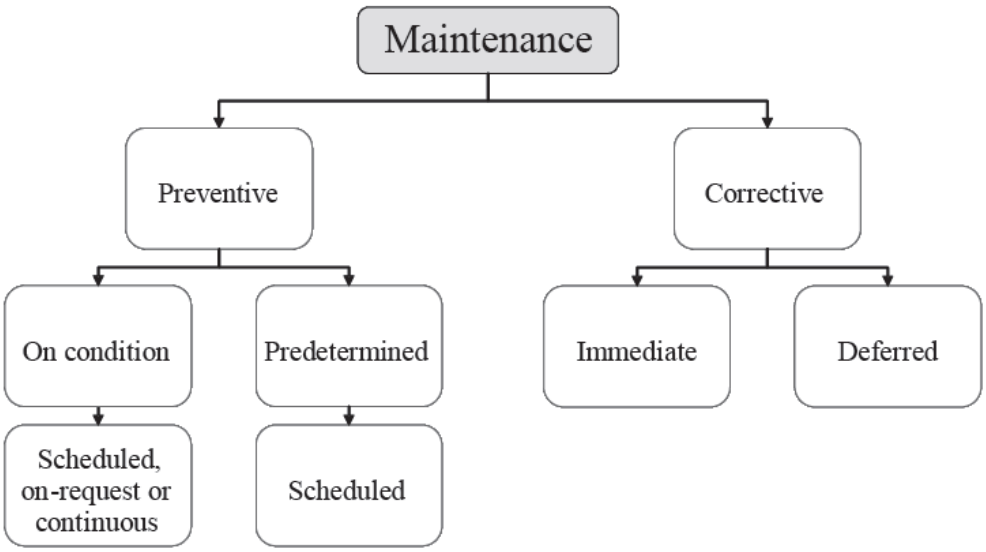


Figure 5. Maintenance types according to standard EN 13306:2001 (Crespo 2007, 70).

2.2.1 Corrective Maintenance

Corrective maintenance, also known as run-to-failure or breakdown maintenance, is carried out to restore or repair an equipment or a system after a fault back to a stage in which it can perform a required function. Corrective maintenance can be immediate or referred based on the timing of maintenance actions. Immediate maintenance is performed without delay after fault detection to avoid unacceptable consequences, whereas deferred maintenance is delayed after fault detection according to given maintenance rules. Figure 6 presents the typical activities of corrective maintenance tasks. (Crespo 2007, 71)

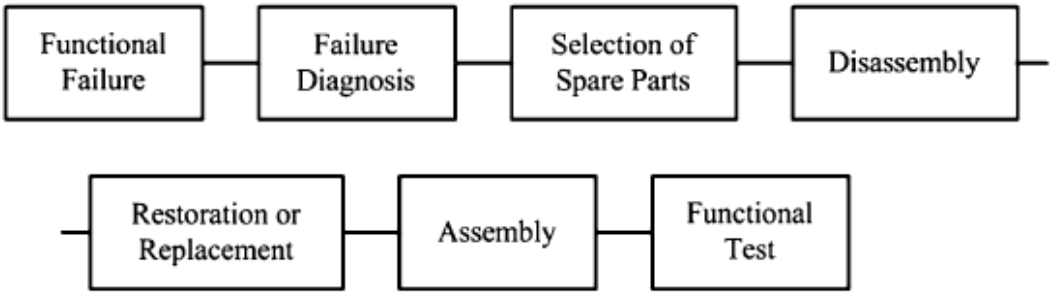


Figure 6. Flowchart of activities of a corrective maintenance task (Martha de Souza 2012, 128).

As corrective maintenance is reactive by nature, it is only appropriate approach if the failure consequences are small. The more complex and critical the component is for the process, the more severe consequences a fault may cause. If applied correctly, corrective maintenance has some advantages, as maintenance planning is simple, and no work is scheduled until needed. Also, some failures are not predictable by any instruments or analysis and manual inspection as part of corrective maintenance is the only way to detect a fault. (Holmberg et al. 2010, 11)

Despite the simplicity and advantages of corrective maintenance, there are some major disadvantages when applied in inappropriate situations. For instance, failures can occur at an inconvenient time leading to loss of production, some faults may go unnoticed causing expensive consequential damage, a large maintenance crew and spares inventory is needed on standby and there is no data available regarding the past, present or possible future state of the equipment. In general, expensive and even dangerous failure consequences should be expected when corrective maintenance is applied inappropriately. (Ibid)

2.2.2 Predetermined Preventive Maintenance

Preventive maintenance aims to maintain equipment in working condition by performing maintenance actions before failures occur to minimize the failure consequences related to corrective maintenance. Preventive maintenance can be predetermined, or condition based. (Crespo 2007, 70–71)

In predetermined preventive maintenance (PPM), maintenance is scheduled in advance based on e.g. time, hours run, operation cycles or production. PPM is used the most effectively, when machine/component life is predictable, or age related. The most important advantage of PPM compared to corrective maintenance, is a more effective use of time, as plant downtime and failure consequences are decreased and maintenance is planned well in advance. PPM also increases machinery's reliability and reduces the need for spares. The typical activities of PPM tasks are presented in Figure 7. (Chanda & Mukhopaddhyay, 2016, 98; Holmberg et al. 2010, 12)

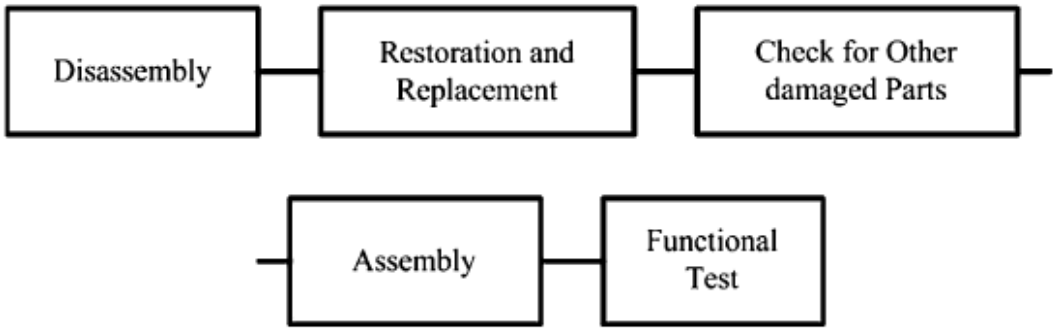


Figure 7. Flowchart of activities of a PPM task (Martha de Souza 2012, 129).

The main disadvantage of PPM is that scheduled actions are often not carried out at optimal time. Equipment may not need maintaining but due to scheduled maintenance actions, spares and labor are used unnecessarily and potential plant downtime is caused. On the other hand, faults may still occur, when components time to failure is shorter than the maintenance interval. Optimizing the maintenance interval to minimize overall costs is crucial to successful PPM. (Holmberg et al. 2010, 12)

2.2.3 Condition based maintenance

In condition based maintenance (CBM), component's health is monitored, and predictive maintenance actions are carried out to prevent failures before they occur. CBM is considered an advanced maintenance strategy combining the benefits of other strategies. The main advantage of CBM is that equipment can be maintained operational with minimum maintenance. This means that better planning of repairs is possible and reduced spare inventory is needed, inconvenient breakdowns and expensive consequential damage are avoided, and the failure rate and unnecessary work are reduced. Studies have shown that implementing a

CBM strategy can reduce the number of component breakdowns by 70% to 75%, maintenance costs by 25% to 30% and equipment downtime by 35% to 45% (Sullivan et al. 2010, 5.4). The typical activities of predetermined preventive tasks are presented in Figure 8. (Holmberg et al. 2010, 15)

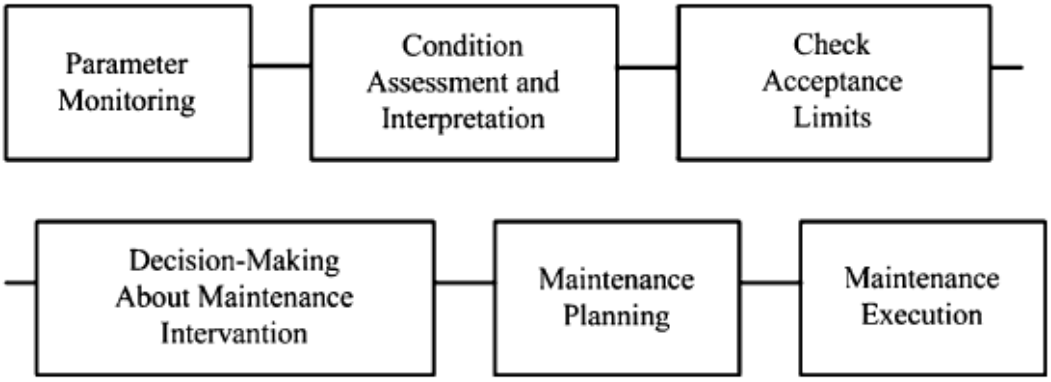


Figure 8. Flowchart of activities of a predictive maintenance task (Martha de Souza 2012, 130).

CBM requires condition monitoring to be effective. Measuring techniques used in condition monitoring can vary from human sensing to sophisticated instrumentation aiming to detect the current state or even to predict the future condition of the monitored component. Different measurement techniques are presented in Section 3. The main challenges of CBM are related to the task of condition monitoring. The required instrumentation and data processing system can be expensive and accurate maintenance can be difficult to achieve due to complexity of the system and its failure mechanisms. (Holmberg et al. 2010, 5; Chanda & Mukhopaddhyay, 2016, 4; Niu et al. 2010, 786–787)

2.3 Characteristics in power plant maintenance

Section 2.2 has introduced the common industrial maintenance strategies summarized in Table 1. Even though condition based maintenance has some superior features compared to less advanced strategies, it is essential to emphasize that the best maintenance policy incorporates all mentioned strategies. Every strategy has their own strengths and limitations and each strategy used appropriately is part of a planned approach. According to Hashemian & Bean (2011), almost 30% of industrial equipment does not benefit from CBM. (Holmberg et al. 2010, 16)

Table 1. Range of maintenance approaches. Adopted from Niu et al. (2010, 787).

Category	Maintenance approaches			
Sub-category	Corrective	Preventive		
	Run-to-failure	Predetermined	Predictive	
	Fix when it breaks	Scheduled maintenance	Condition based maintenance diagnostics	Condition based prognostics
When scheduled	No scheduled maintenance	Based on a fixed time schedule	Based on current condition	Based on forecasting of remaining equipment life
Why scheduled	N/A	Intolerable failure effect and possibility of preventing the failure effect	Based on evidence of needs	Maintenance need is projected as probable within mission time
How scheduled	N/A	Based on the useful life of the component forecasted during design and updated trough experience	Continuous collection of condition monitoring data	Forecasting of remaining equipment life based on actual stress loading
Kind of prediction	None	None	On and off-line, near real-time trend analysis	On and off-line, real time trend analysis

There are different models and guidelines to maintenance management and how to assign maintenance strategies on different industrial fields, one popular approach being reliability centered maintenance. The generally applied maintenance strategies are also used in thermal power plants but compared to traditional maintenance practices, thermal power plants have some characteristic features. First, power plants are most often run continuously, and maintenance actions may cause outages of the plant operation. Duration of outages are restricted by market demand and therefore, an appropriate maintenance philosophy with a short-term, medium-term and long-term approach to the market demand pattern is required. Second, outages during maintenance cause loss of production, which is often a major expense with

unplanned outages. An example of unplanned unavailability costs is presented in Figure 9. Elaborated further, commercial earning of a power generator is dependent on the grid availability and operating efficiency, and therefore, availability is often more valued criterion than reliability. Third characteristic feature of power plant maintenance is expensive main capital equipment, which means that some key process equipment cannot be duplicated that in turn leads to limited redundancy. (Chanda & Mukhopaddhyay, 2016, 4; Ndjenja & Visser, 2015, 136)

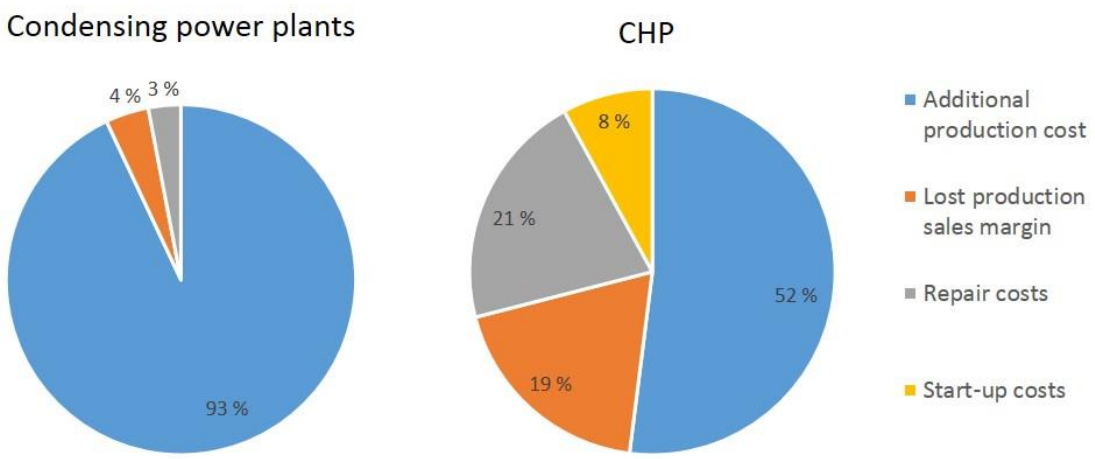


Figure 9. Unplanned unavailability costs in power plants. Adopted from Kylliäinen et al. (2011, 4).

Most modern power plants use a combination of corrective and preventive maintenance strategies. The most effective strategy for individual items and components are determined by several factors, e.g. the component's failure rate and criticality, which means that the optimal combination of maintenance strategies is unique for a specific plant or system to enhance the plant's availability at the lowest maintenance cost. (Chanda & Mukhopaddhyay, 2016, 4; Ndjenja & Visser, 2015, 136)

The use and continuing development of various diagnostic techniques, high system and component automation level and more efficient work organizations have contributed to the better optimization of maintenance practices in recent years. Despite continuous improvement and development, a lot of potential for further cost reduction still exists, especially by extending

condition monitoring to equipment not rated critical. Minimizing maintenance costs is critical for power plant operators, since maintenance can cover up to 30% of all electricity generating costs. (Gräber 2004, 1–2)

3 CONDITION MONITORING

Condition monitoring is a process of monitoring parameters that provide information about changes in the component's or system's condition or performance. Condition monitoring can be applied online to assist the continuous operation or as an offline service to support longer term operation. Condition monitoring includes all the actions from data acquisition to estimating the state of health of the monitored component. A generalized condition monitoring framework is illustrated in Figure 10, with each box presenting one device. The first box contains data acquisition device, whose primary function is to acquire data from the system. Data acquisition devices usually include measurement devices, such as thermometers, accelerometers or strain gauges. The second component is data preprocessing, which can be integrated to data acquisition or treated separately. Often some preprocessing is integrated to data acquisition to reduce the data transfer and storing requirements, and additional preprocessing is performed with separate modules to enable efficient data analysis. Some data preprocessing techniques are described in Section 4. The preprocessed data are processed further in feature selection device, which is a process of identifying and quantifying specific aspects of the data that are proper indicators of faults in the structure or process. In the decision making device, the selected features are interpreted to obtain condition diagnosis as the outcome. (Marwala 2012, 4–5)

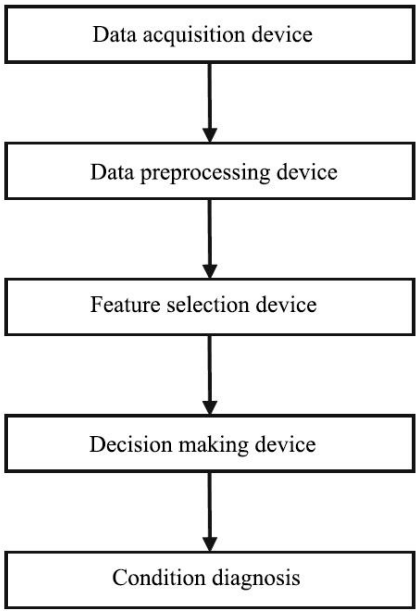


Figure 10. Condition monitoring framework. Adopted from Marwala (2012, 4).

3.1 Fault detection, diagnosis and prognosis

Three important stages in condition monitoring can be identified: fault detection, fault diagnosis and prognosis. Fault detection is a process to recognize abnormal operating conditions indicating faults in the monitored system. Diagnosis in turn, is a process to analyze the existence and the cause of the problem, whose objective is to examine symptoms and syndromes to determine the nature of faults and failures (ISO 13372, 2012). Diagnosis is comprised of fault isolation, which is a task to locate the faulty component, and of fault identification, which is a task to determine the nature and severity of the fault. The definition of diagnosis varies in the literature. Fault detection can be treated separately, as in this thesis, or it can be seen as part of diagnosis. In practice, fault identification phase rarely appears, and thus, the common abbreviation used in many scientific papers is FDI (Fault Detection and Isolation) (Patan 2008 ,10). (Van Tung & Yang, 2009, 62; Jardine et al. 2006, 1485)

The final and most appealing stage of condition monitoring process is prognosis, which is a process of predicting a future condition from present signs and symptoms. Prognosis aims to predict faults before they occur determining the failure mode, when the fault will occur and its chance of happening (Leturiondo 2016, 15). Prognosis predicts the component's time

to failure in terms of Remaining Useful Life (RUL) within design parameters. RUL, sometimes referred as residual service life, can be defined as the time left until the monitored unit cannot perform its specified function given the current machine age and condition, and the past operation profile. RUL can be expressed as conditional random variable as:

$$\mathrm{RUL} = T_{\mathrm{f}} - t|_{T_{\mathrm{f}}>t,Y(t)}, \tag{1}$$

where T_{f} denotes the random variable for time to failure, t is the current age and $Y(t)$ is the past condition profile up to the current time. Being a random variable, the information obtained from Eq. 1 can be divided into two characteristics: Estimation of the RUL and probability distribution of the RUL. When in healthy state, statistical estimations can be used to calculate the RUL but when evidence of a failure occur, accurate information from the diagnosis process is required for the RUL estimation. (Leturiondo 2016, 14–18; Yan 2014, 153–154; Pusey 2007, 3–4)

Stages of condition monitoring to support maintenance decision making are presented in Figure 11. However, it should be highlighted that fault diagnosis and prognosis require clear differentiation between different faults, which in turn requires a fault library of the possible faults of the monitored item. Creating and sustaining a fault library is possible when monitoring a limited amount of equipment with known fault mechanisms. However, extending the fault library to cover plant-wide monitoring is still practically impossible. In plant-wide monitoring, the main aim of condition monitoring is still to detect fault development as early as possible and assist human decision making.

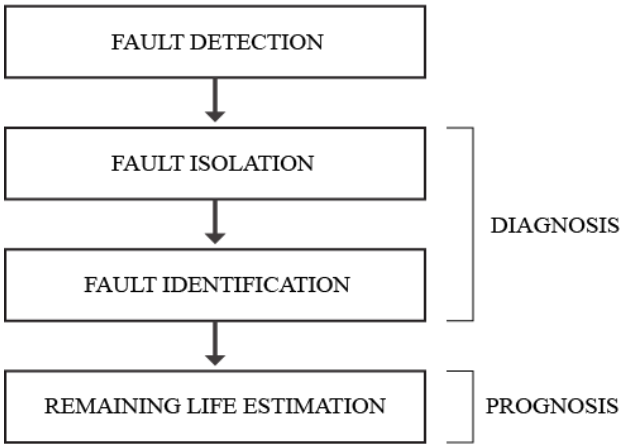


Figure 11. Stages of condition monitoring to support maintenance decision making.

3.2 Condition monitoring measurement techniques

Several measurement techniques have been applied in condition monitoring aiming at providing information of the actual condition of the monitored component or system. The simplest form of condition monitoring are routine inspections carried out by plant personnel to detect local abnormalities, such as noise, smell or vibration in the running component. While the importance of inspections by experienced personnel cannot be neglected, more accurate automatized and intelligent condition monitoring systems are desired, especially for decentralized production and remote support services. Some of the most common sensing techniques are vibration monitoring, lubricant analysis, wear particle analysis, monitoring of performance parameters and deterioration monitoring using non-destructive techniques. (Manning-Ohren 2015; Martha de Souza 2012, 130–131; Marwala 2012, 1)

3.2.1 Vibration monitoring

All running machines vibrate while running. Changes in vibration levels and patterns can be used for detecting faults and identifying the health of an equipment. Vibration monitoring is the most used measurement technique in condition monitoring, especially with rotating machines such as pumps, compressors, turbines and electrical generators and motors. Typical measured parameters are acceleration, velocity and displacement of moving mechanical components. These parameters can be measured directly at the component of interest but more often vibration is measured at an available surface giving an indication of internal events without disrupting the process. Vibration monitoring equipment consists of three main items: transducers for data acquisition, signal processing system and an algorithm for condition assessment. Typically, the raw vibration data is processed into overall vibration levels, frequency spectra or high frequency emission. (Opetushallitus; Martha de Souza 2012, 131; Holmberg et al. 2010, 20)

3.2.2 Lubricant analysis and wear particle analysis

Lubricant analysis measures the additives, contaminants and debris, which reveal the condition of the lubricant. The analysis aims at scheduling lubricant change intervals optimally to maintain satisfactory equipment operation. Lubricant analysis is often coupled with wear particle analysis when samples of lubricant oil are collected and analyzed in pre-determined intervals. Rate change of debris collection indicates changes in the system condition. Wear

particle analysis is also important to avoid high level of wear particles from rubbing surfaces in the machinery. (Martha de Souza 2012, 131; Holmberg et al. 2010, 21)

3.2.3 Non-destructive deterioration monitoring

Several non-destructive techniques, which evaluate the properties of a material without affecting the integrity of the component under test, are being used to monitor deterioration of some components. Typically, non-destructive testing is used for static equipment, such as pressure vessels, piping systems and structural components which undergo high stresses due to high temperature and pressure or are prone to corrosion and erosion. Non-destructive techniques allow inspections of difficult-to-reach hot spot stress areas without interfering the process. It is also possible to define the importance of detected defects for structural integrity to define the need for maintenance actions. The most used non-destructive techniques include magnetic particle inspection, eddy current inspection, acoustic emission testing, radiographic inspection and ultrasonic inspection. Thermographic inspection, which measures infrared energy emissions, is also considered a non-destructive technique but is usually used to evaluate possible cumulative damage evolution in electrical equipment. (Martha de Souza 2012, 131; Urata et al. 2015, 37–39)

3.2.4 Performance monitoring

Monitoring the performance of an equipment is an important aspect of condition monitoring. The performance, which can be inspected by monitoring performance or process parameters, such as pressure, flow rate or energy consumption of an equipment, can indicate the condition of a machine. These indications can be used to diagnose the system operational condition. Failure condition can be recognized using fixed limit values or by detecting abnormalities. Traditionally, experienced personnel are required to detect and identify failure modes from process parameters, but the recent development of computational methods have led to a point, where the need for human diagnosis and localized inspection techniques can be greatly reduced using computational methods. Computational intelligence methods, such as artificial neural networks, have gained a great deal of attention and led to the development of global methods, which can use changes in measured data as a basis for fault detection reducing human effort and detecting faults at early states invisible to traditional approaches. (Marwala 2012, 1; Holmberg et al. 2010, 20; Martha de Souza 2012, 131)

4 DATA PREPROCESSING

Data acquisition, which is a process of collecting and storing useful data from the monitored asset, is a crucial part of any condition monitoring system both for monitoring and modelling purposes. The collected data under a CBM program can be categorized in two main types: event data and condition monitoring data. Event data include the information of performed maintenance actions, e.g. installations and repairs, and is usually added manually to the system. Condition monitoring data consist of the data formed from measurements related to the process or the condition of the component. The acquisition of condition monitoring data is usually integrated to plant automation. The process data can be augmented with additional instrumentation, e.g. portable devices, on the most critical components, but extending additional instrumentation to cover the whole plant is often too expensive for CBM purposes. Therefore, methods capable of detecting fault development from process data without the need for additional instrumentation are desirable to extend accurate condition monitoring to cover as many equipment as possible. As described in Section 3.2, different measurements are used in condition monitoring and therefore, condition monitoring data are very versatile. The data can be, for example, vibration data, acoustic data, oil analysis data, temperature, pressure or environment data. The instrumentation requirements also vary between machinery. Some slowly changing parameters, e.g. temperature, can be measured with an interval of seconds or even ten seconds, whereas fast parameters, e.g. vibration, are usually measured with a period of milliseconds. The measurement of fast changing parameters can be then averaged over a longer time period depending on the data analysis requirements. (Niu 2017, 50; Jardine et al. 2006, 1485; Kaija 2015, 25–26)

Data preprocessing includes any type of raw data processing to enhance the reliability of the data, which in turn improves the accuracy of data analysis (Niu 2017, 71). Data preprocessing methods can be broadly divided into two categories: data preparation and data reduction. Data preparation includes methods and techniques that are used to initialize raw data in a way that the data can be effectively processed for the purpose of the user, for example, in a modeling software or for expert analysis. Data reduction refers to a set of techniques, which obtain a reduced representation of the data making analysis algorithms faster and more feasible. Data reduction is not always mandatory for CMB purposes, but it can be

highly useful for machine learning algorithms and when using large amounts of data. (Garcia et al. 2015, 11–13; Bangert 2012, 37–38)

The type and extent of required data preprocessing greatly varies between applications and the chosen approach. Many methods also overlap and can serve several purposes, e.g. data cleaning and transformation, at once. The most used methods include (Bengtsson et al. 2004; Niu 2017, 71):

Data cleaning to remove noise from data,

Sampling to select a representative subset from a large population of data,

Transformation to manipulate raw data into a more convenient format

Feature selection/extraction to extract specified data significant in some particular context.

4.1 Data cleaning

The first step of data preprocessing is data cleaning. Sensors need to operate under harsh conditions in power plant environment, which may lead to unwanted effects and so-called dirty data not representing the actual measured quantity. The cause for unreliable dirty data can be, for example, broken or faulty sensor due to clogging, dirt or overheating. Also, incorrect sensor installations, various temporal and programming effects and sources of noise along the measurement chain can cause artificial variation to the measurements. Since event data is often entered manually to the system, also event data contain errors due to human factor. (Stefaniak et al. 2014, 200–203; Bangert 2012, 37–38; Jardine et al. 2006, 1485)

Outlier detection is one form of data smoothing and noise reduction carried out to capture the studied dynamics. Outliers can be detected based on their type. Unrealistic data e.g. temperature below 0 Kelvin are eliminated by introducing a minimum and maximum allowed value for every sensor. Unlikely data, which are not necessarily unphysical but most likely caused by faulty sensors can be detected with statistical tools, usually setting acceptance limits in terms of number of standard deviations away from the mean. In contrast to fixed limits, statistic parameters change over time creating dynamic limits to accepted values. Third method to detect outliers is clustering. In this method, clusters of data points presenting a particular operating condition are formed. Data points not belonging to any cluster are

irregular by definition and can be excluded from processing. Outlier detection is also a problem of missing data. Often missing data points can be replaced with artificial values, obtained with interpolation or with soft sensors, to enable data analysis. Other possibility is to exclude the missing data from the analysis, but this may be problematic to the analysis, especially if the missing data have a structured cause. (Bangert 2012, 42)

As with other data preprocessing methods, also data cleaning methods vary between applications. One particularly important field is signal processing, which refers to data processing for waveform data. Waveform data, e.g. vibration signals, are widely used in condition monitoring applications. Several techniques to de-noise, i.e. to increase the signal-to-noise ratio, waveform data are used, e.g. wavelet transform and different filters that remove portions of the signal identified as noise. (Niu 2017, 70–71; Roulias et al. 2013, 1045–1046; Bangert 2012, 47)

4.2 Sampling

Sampling is a data reduction technique used to ease the data analysis of large data sets. For data-driven modeling, sampling refers to the decision of which data points are presented to the learning algorithm. Sampling also includes the decision, whether the data points are presented once (sampling without replacement) or multiple times (sampling with replacement) to represent a certain condition. Besides reducing the amount of used data, data sampling is also important to balance the data regarding the occurrence of rare events. Data sampling can have several forms, random sampling and stratified sampling being the two basic techniques. In random sampling, data points are chosen based on a probability distribution, which is usually the uniform distribution resulting in equal probability for all data points. Stratified sampling is used with imbalanced data sets selecting the sample data points based on a probability distribution that results in over-sampling rare data points representing abnormal conditions, and in under-sampling common data points representing normal frequently occurring conditions. (Bangert 2012, 50–51; Garcia et al. 2015, 156–158)

4.3 Data transformation

Data transformation is a process of converting raw data in a way that it can be applied more efficiently (Garcia et al. 2015, 12). The type and extent of data transformation is highly dependent on the data type and on the analysis method. For instance, vibration signals by default require more preprocessing, including wavelet transformation and Fourier transformation, in comparison to temperature data. Data transformation includes several methods, e.g. smoothing, feature construction and aggregation, discretization and normalization being the most important and used ones.

Discretization transforms continuous attributes into discrete ones (Garcia et al. 2015, 244). In industrial setting, most of the sensor readings are identical. As storing every single measured value is often inefficient, the time-series from instrumentation is discretized and only certain values accepted as sufficiently new are recorded. Usually the discretization is done according to one of the three following criteria: a new value is recorded at a manually set fixed time interval; a new value is recorded if it differs enough from the previously recorded value or a new value is recorded at a fixed time interval, unless the value is outside certain boundaries when the rate for value recording is adjusted to keep a detailed record of unusual events. (Bangert 2012, 38–40)

Normalization refers to data transformation techniques that do not generate new attributes but transform the distribution of the original values resulting in a new set of values. Normalization is a standard data preparation method for analysis based on statistical methods and machine learning. The most used normalization technique is the min-max normalization scaling all numerical values to a specified range, usually from 0 to 1. Other common techniques are Z-score normalization that uses standardization based on standard deviation, and decimal scaling normalization that uses power of then division to transform a numerical attribute resulting in absolute values lower than 1. (Garcia et al. 2015, 46–48).

4.4 Feature selection/extraction

Feature selection is a common technique for data and dimensionality reduction. Feature selection is used to identify irrelevant or redundant features to enable more efficient classifi-

cation. Feature selection algorithms fall into three categories: filters are used to extract features from the data without any learning involved; wrappers use learning techniques to find useful features; and embedded techniques which combine feature selection and classifier construction. (Hira & Gillies, 2015, 1–3; 2; Bania 2014, 3–4)

Feature extraction is also used in data reduction. The difference compared to feature selection is that feature extraction calculates new features from the original ones. In general, feature extraction is a mapping of a multidimensional space into a lower-dimensional subspace. The most well-known technique for dimensionality reduction is principal component analysis described in Section 5.2.1. (Hira & Gillies, 2015, 6–7)

5 MODELLING METHODS OF CONDITION MONITORING

Applying CBM strategy to thermal power plants requires system modelling and process monitoring. The main approaches to system modelling are physics-based modelling based on a prior knowledge, data-driven modelling, and hybrid modelling, which combines several modelling techniques (Leturiondo 2016, 17). A classification of modeling methods in condition monitoring is presented in Figure 12 and these approaches are discussed in this section. Numerous models for fault detection, diagnosis and prognosis have been reported in the literature to support maintenance decision making. As the field of condition monitoring modeling is enormous, this section is not intended as a comprehensive list of modelling methods but rather to provide an overview of different approaches. This section concentrates on fault detection, since diagnosis and prognosis are still very difficult to apply in plant-wide monitoring.

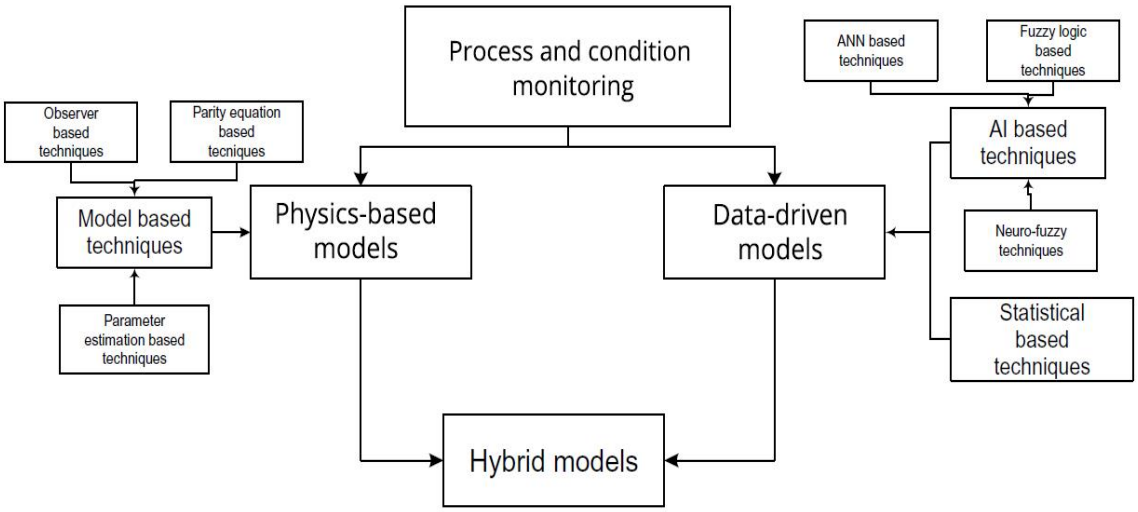


Figure 12. Classification of system modeling approaches in process and condition monitoring. Adopted from Das et al. (2012, 722).

In the context of CBM, the main task of modelling is to detect abnormalities indicating faults and malfunctions. In the simplest form, fixed limits are assigned to each monitored measurement and alarms are created when measurements exceed the threshold. Although simple, this approach has multiple limitations. Manually applied limits require a great deal of process knowledge and engineering work but the limits are still mainly based on assumptions and experience. Fixed limits are also difficult to set in a way that they are representative for changing operation and environmental conditions. The main technique to overcome these limitations is to use residuals as fault detector. Deviations between the monitored and expected values, called residuals, are artificial time-varying signals reflecting potential faults, see Figure 13. In healthy state, the residual is normally designed to be zero and significantly differ from zero when the monitored item is in faulty state. As real systems are always subject to noise and models possess some level of uncertainty, the residual value is never expected to be zero but rather in the neighborhood of zero. (Frisk 2001, 9–11; Venkatasubramanian et al. 2003a, 300–302)

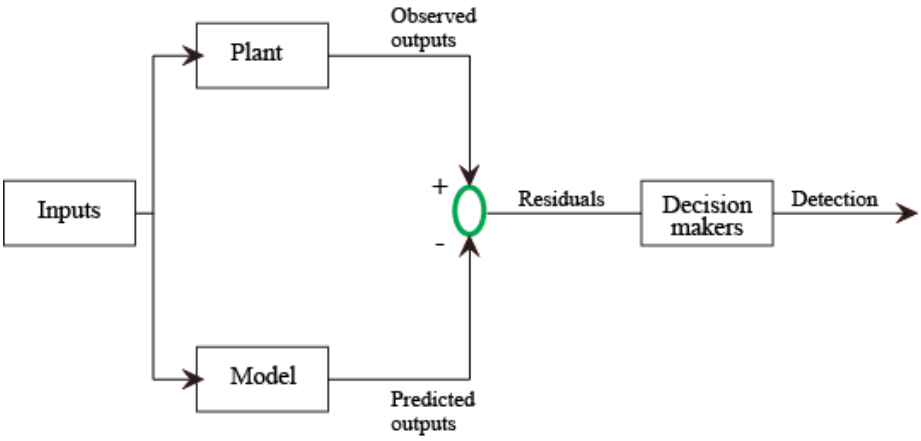


Figure 13. Schematic illustration of fault detection using residuals. Adopted from Sheriff et al. (2017, 238).

The residual approach consists of residual generation and residual evaluation. Residual generation refers to the comparison of measured and predicted outputs obtained by the chosen model, and residual evaluation to the analysis whether a fault has occurred including also fault isolation (Patan 2008,10). Residual generation is considered a more important stage, since residual evaluation is fairly simple if residuals are well-generated (Sobhani-Tehrani & Khorasani, 2009, 12). In many applications, residual generation is enough for fault detection, when residuals correspond directly to the equipment under abnormal behavior or the main task of condition monitoring is merely to get the operator's attention (Stanley 2010). For fault isolation and diagnosis, residual evaluation including a prior knowledge of possible faults is required (Frisk 2001, 12).

5.1 Physics-based modeling

Physics-based modeling, also known as modeling by first principles, is based upon natural laws providing a quantitative description of the system and its physical phenomena (Stringer et al. 2012, 156). Physics-based models can be classified in various ways. For instance, whether the model considers time variations, models are classified as static or dynamic or as linear or non-linear based on the obtained set of equations. (Jaako 1998, 9–12)

Static models present the modeled process at a single point in time and cannot describe time variations. These models are described by algebraic or partial different equations where the time derivative term is neglected. This makes them simpler and faster to evaluate than dy-

namic models, but since real processes are hardly ever static, the applicability of static models is limited mostly to process design and to slowly changing processes, which can be approximated as non-time-dependent. Dynamic models, in contrast to static models, describe time variations. These models are described by ordinary differential equations, differential algebraic equations or partial differential equations making them more complex to solve than static models. (Jaako 1998, 10–11; Ge 2017, 21)

One classification of physics-based models used in condition monitoring applications is to classify the models as system models or physics-of-failure (PoF) models. System models rely on residuals, whereas the PoF approach uses damage models to determine fault generation and propagation (Pecht & Gu, 2009, 310). This approach requires an accurate analytical model, which describes how variables like material properties, geometry, load cycles and operating conditions impact on different failure modes and degradation mechanisms. PoF models are widely used in electronic systems (Temsamani et al. 2017). In power generation, they are mostly implemented to components, such as piping systems (Di Maio et al. 2015; Chookah et al. 2011) and turbine discs (Zhu et al. 2016), object to creep, fatigue, erosion and corrosion. These models are highly specific to each component and application and are mainly used in reliability assessment, but they can also be implemented into more fast-paced condition monitoring to provide continuously updated predictions of the structural health of some critical components (Tautz-Weinert & Watson, 2017). (Pecht & Jaai, 2010, 318; Martha de Souza 2012, 261–262)

Another classification of physics-based models used in condition monitoring is based on residual generation and evaluation approaches, as presented in Figure 12. The most used techniques fall into three categories: parameter estimation based techniques, observer based techniques and parity equation based techniques. The main idea of comparing measured and predicted values is the same for all three approaches, but the mathematical presentations of predicted values differ.

Parameter estimation techniques rely on detecting changes in the physical process parameters not considered in the development of the model. A physical process parameter can be, for example, heat transfer coefficient in the case of heat exchangers. The model parameters

are functionally related to the physical parameters and this relationship in the form of mathematical equations is used to estimate the physical parameters. The estimated physical parameters can then be compared to measured values. Observer, or filter, based approach monitors a process with state variables. Observer based methods compare measurements to model based reconstructions of the system outputs that are estimated by different observers and filters, e.g. Kalman filters. Parity equations are rearranged and often transformed variants of the input–output or state–space model of the considered process. Parity equations are used to estimate the residual values. Rearranging the model structure enables efficient fault isolation by generating residual vectors orthogonal to each other for different faults. (Das et al. 2012, 734–736; Tehrani & Khorasani, 2009, 12–13; Patan 2008, 11–14)

The first step of physics-based modelling is always simplifying the considered process. This is achieved with assumptions to simplify the calculations or to enable them at all with an acceptable expenditure. After simplifying assumptions, the description of the system is formulated in the form of either ordinary or partial differential equations, which can be classified as balance equations, physical or chemical equations of state, phenomenological equations or interconnection equations. Defining the system model often leads to complex equations, which can be simplified with mathematical treatments like linearization, approximation with lumped parameters and order reductions, before solving the obtained equations using numerical methods. (Leturiondo 2016, 17; Isermann 2006, 71)

Physics-based models stem from natural laws, and as such, they have several advantages over other approaches. Physics-based models are general models and can produce a very precise description of the studied system and phenomena. These models do not depend on comparisons to experimental data, making them not bound to certain environmental or operational conditions. This does not only make physics-based models applicable to different situations but also allows simulating new insertions to the system and produce results for analysis. (Leturiondo 2016, 17; Isermann 2006, 71; Stringer et al. 2012, 156)

The downside of using physics-based models is their complexity. Creating the model is always balancing between accuracy and complexity and the associated cost and time. Creating an extensive and accurate physics-based model can be practically impossible or the model

can become excessively complex, and despite the rapid development of computational power, the calculation process too demanding for practical purposes. Therefore, simplification of the modeled system, the model itself and the solved equation set is required to obtain a technically feasible model. (Jardine et al. 2006, 1494; Stanley 2010)

5.2 Data-driven modelling

Unlike in physics-based modelling, data-driven models are not based on natural laws or physics of the system. Instead, data-driven models are constructed from measurements and rely on monitored and historical data that are used to train the system behavior to the model. Since data-driven models have no causal relationship between the model inputs and outputs, they are also referred as black-box models, whose reliability must be questioned. On the other hand, data-driven models do not require a priori knowledge of the underlying physics of the system and can thus be implemented to complex systems, where physics-based approach is not feasible. (Leturiondo 2016, 18; Sutharssan et al. 2015, 2)

Data-driven models are based on the assumption that the statistical characteristics of the system do not change until a fault occurs (Pecht & Jaai, 2010, 318). This makes data-driven models faster to implement and computationally more efficient than physics-based models. The main disadvantage of data-driven models is the absolute dependency on historical and empirical data, making these models applicable only to operating conditions in which historical data is available. Inappropriate dataset selection for the model construction can lead to false alarms (type I error) or inability to detect process faults (type II error). For reliable diagnosis and prognosis, also run-to-failure data is required, which are often not available for a particular system or component. Run-to-failure data sets are also used to generate respective threshold values for monitoring purposes. Even though these data sets can be difficult to obtain, for instance to critical and expensive components or to new systems without any historical data, they can be achieved with several techniques. These techniques can be e.g. hardware-in-the-loop simulations, where the hardware is put under simulated loads as in the real application or accelerated life test, where the component is put under elevated stress to fail more quickly. (Ge 2017, 18; Sutharssan et al. 2015, 2)

Data-driven approaches can be classified as statistical approach or computational intelligence (CI) approach. Section 5.2.1 describes the characteristics of statistical models and some standard methods. Computational intelligence is a branch of artificial intelligence (AI) widely used in different fields of research and engineering practice. Computational intelligence uses computational methods inspired by processed found in nature and in the human brain being one of the most effective data processing tools to solve complex nonlinear problems. Two of the most important CI methods used in condition monitoring applications are artificial neural networks, which are discussed in Section 5.2.2, and approaches based on fuzzy logic. (Zhang et al. 2017, 783; Das et al. 2012, 728–730)

A fuzzy logic system (FLS) is generally a nonlinear mapping of an input data vector into a scalar output (Mendel 1995, 346). The process consists of four components presented in Figure 14: fuzzifier, rulebase, interference engine, and defuzzifier. The first component is used to fuzzify crisp input data to apply linguistic statements based on expert knowledge in the form of IF-THEN rules. These rules are combined and applied in the interference engine, which mimics the human decision making based on fuzzy concepts. The mapped fuzzy output sets are finally defuzzified to produce a crisp output. (Song & Johns, 1997, 220–222)

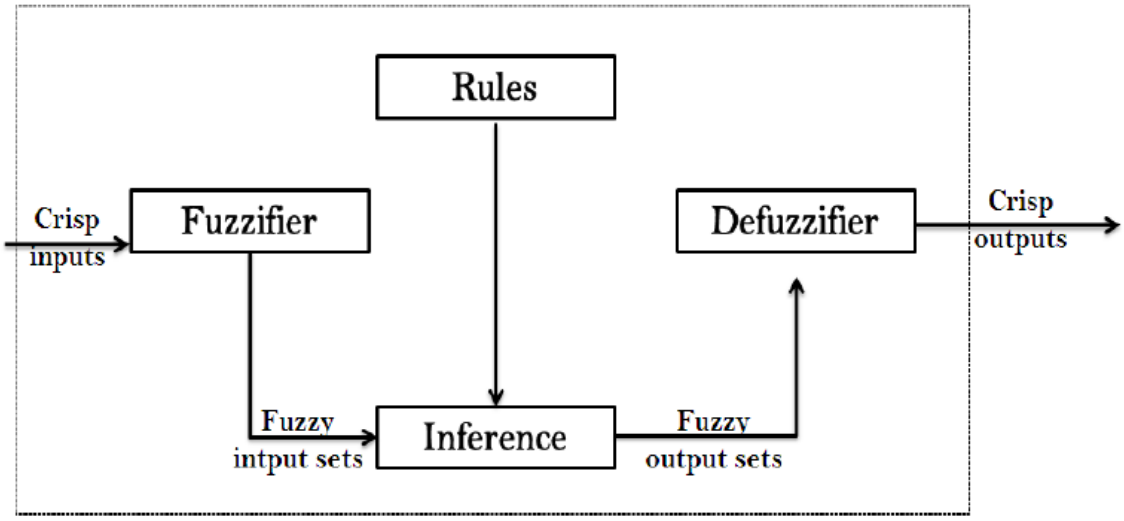


Figure 14. Block diagram of a FLS (Largueh et al. 2015).

Fuzzy logic systems can effectively use inexact, subjective and ambiguous data and vague knowledge elements giving the ability to map realistic situations. FLS imitates logical human thinking, making FLS applicable for qualitative analysis of complex large-scale systems and is therefore used widely in many different industrial fields. In condition monitoring, FLS are often used in conjunction with other methods to aid in the decision making process, mainly due to slow interference speed and low precision. (Zhang et al. 2017, 784–785, 787–788; Suganthi et al. 2015, 586–587; Shanmuganathan 2016, 12)

5.2.1 Statistical models

Statistical approaches to condition monitoring and fault detection are the oldest and most widely employed strategies partly due to their easy development and implementation. Most statistical approaches are built from normal operational data to represent normal healthy operation. Analysis of the underlying statistical properties, such as variance and standard deviation of the monitored data, is carried out to detect changes in these parameters. Statistical approaches can be divided into univariate approach and into multivariate approach. Techniques under the former category monitor each characteristic independently, which makes them impractical to deploy in modern industrial applications. Multivariate approaches do not face the same challenge, since they monitor simultaneously many characteristics and also take into account the interdependencies existing among these characteristics. (Das et al. 2012, 725; Sutharssan et al. 2015, 3; Severson et al. 2016, 191; Qin 2012, 221)

Multivariate techniques are powerful tools whose primary function is to convert a number of related process variables to a smaller set of uncorrelated values. Multivariate techniques have the ability to compress data and reduce its dimensionality while preserving essential information making the analysis easier compared to the original larger data set. They are also capable of handling noisy data and correlation to effectively extract true information. Many techniques have been researched and implemented in numerous areas and industrial fields, two of the most used techniques being principal component analysis (PCA) and partial least squares (PLS). (Venkatasubramanian et al. 2003b ,331)

Principal component analysis is a dimensional reduction technique based on orthogonal decomposition of the covariance matrix of the process variables along directions that explain the maximum variation of the data. Dimensionality is lowered by finding factors having

lower dimension than the original data set and the ability to properly describe the major trends in the original data set. (Venkatasubramanian et al. 2003b ,331)

The PCA model can be illustrated as presented by Sheriff et al. (2017, 239–241) and shown in Figure 15, where **X** denotes input data matrix with dimensions of n x m, n being the number of process variables and m the number of observations. Using single value decomposition, the input data matrix can be expressed as

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathrm{T}} \tag{2}$$

where **T** is a matrix of the transformed variables and **P** is a matrix of the orthogonal vectors. The columns of the matrix **P** are eigenvectors, also known as loading vectors, associated with the covariance matrix of the input data matrix **X**, which is given by

$$\sum = \frac{1}{n-1}\mathbf{X}^{\mathrm{T}}\mathbf{X} = \mathbf{P}\mathbf{L}\mathbf{P}^{\mathbf{T}} \text{ with } \mathbf{P}\mathbf{P}^{\mathrm{T}} = \mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{I}_m \tag{3}$$

where **L** is a diagonal matrix related to the m principal components and $\mathbf{I}_m$ is the identity matrix. The number of principal components of the PCA model is the same as the number of original process variables in the input data matrix, but since most industrial process variables are highly correlated, the process variations can be captured with fewer principal components. The PCA model quality is determined by the number of obtained principal components, too many leading to noise and too few to decreased prediction ability. Selection of the number of principal components is therefore crucial in the model creation. Various techniques, e.g. cumulative percent variance and cross validation, can be used to select appropriate number of principal components. In practice, usually the first two or three principal components are enough to capture the variability and thus, the dimensionality is significantly reduced. (Sheriff et al. 2017, 239–241; Venkatasubramanian et al. 2003b, 331)

Finally, the input data matrix can be expressed as

$$\mathbf{X} = \widehat{\mathbf{T}}\widehat{\mathbf{P}}^{\mathrm{T}} + \widetilde{\mathbf{P}}\widetilde{\mathbf{T}}^{\mathrm{T}} = \overbrace{\mathbf{X}\widehat{\mathbf{P}}\widehat{\mathbf{P}}^{\mathrm{T}}}^{\hat{\mathbf{X}}} + \overbrace{\mathbf{X}(\mathbf{I}_m - \widehat{\mathbf{P}}\widehat{\mathbf{P}}^{\mathrm{T}})}^{\mathbf{E}} \tag{4}$$

where $\widehat{\mathbf{T}} \in \mathrm{R}^{n \times l}$ denotes the matrix of the l retained principal components and $\widetilde{\mathbf{T}} \in \mathrm{R}^{n \times m-l}$ the matrix of the ignored $(m-l)$ principal components. Similarly, $\widehat{\mathbf{P}} \in \mathrm{R}^{m \times l}$ and $\widetilde{\mathbf{P}} \in \mathrm{R}^{m \times m-l}$ denote the matrices containing the l retained eigenvectors and the ignored $(m-l)$ eigenvectors, respectively. Equation 4 expresses the original input data matrix with the modeled variation $\widehat{\mathbf{X}}$ using only the l retained principal components and with matrix **E** representing residuals formed by variations corresponding to process noise. (Sheriff et al. 2017, 240–241; Mansouri et al. 2016, 2)

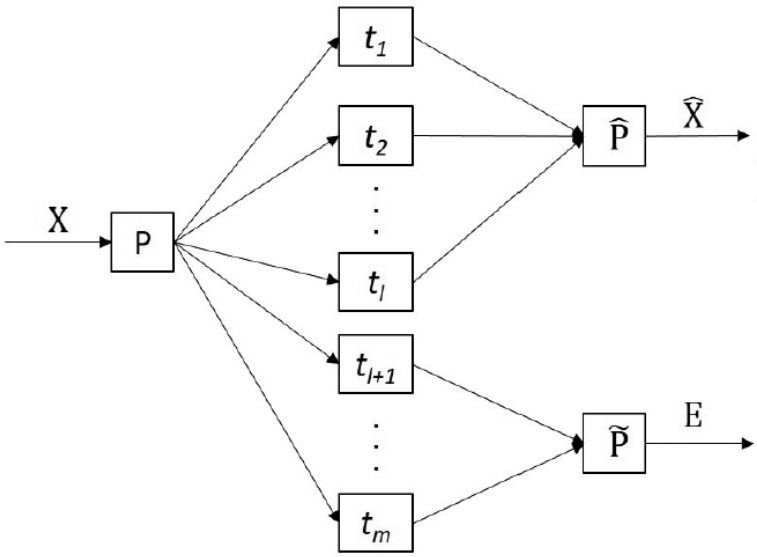


Figure 15. Schematic illustration of PCA (Sheriff et al. 2017, 241).

Partial least squares is another standard linear dimensionality reduction technique. Similarly to PCA, PLS can be used with ill-conditioned data. The main advantage of PLS is that it considers two sets of variables: predictor variables **X** and response variables **Y**. The former represent process characteristics while the latter represent quality characteristics. PLS can evaluate the relationship between **X** and **Y** maximizing the covariance between them in the reduced space defined by the number of latent variables and can be used as a regression tool to predict the quality characteristics from the corresponding process characteristics. PLS does not have a closed-form solution and thus uses an iterative algorithm. (Das et al. 2012, 726–727; Severson et al. 2016, 192; Tidriri et al. 2016, 66)

One standard algorithm, nonlinear iterative partial least square (NIPALS), can be summarized as presented by Sheriff et al. (2017, 241–243), as follows. Score matrices and loading vectors are calculated as:

$$\mathbf{X} = \mathbf{S}\mathbf{P}^{\mathrm{T}} + \mathbf{E} = \sum_{i=1}^{M} \boldsymbol{s}_i \boldsymbol{p}_i^{\mathrm{T}} + \mathbf{E} \tag{5}$$

$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^{\mathrm{T}} + \mathbf{F} = \sum_{i=1}^{M} \boldsymbol{u}_j \boldsymbol{q}_j^{\mathrm{T}} + \mathbf{F} \tag{6}$$

where $\mathbf{S} \in \mathrm{R}^{n \times M}$ is the orthonormal input score matrix and $\mathbf{U} \in \mathrm{R}^{n \times M}$ the orthonormal output score matrix; $\mathbf{E} \in \mathrm{R}^{n \times m}$ and $\mathbf{F} \in \mathrm{R}^{n \times \mathrm{p}}$ are the model residues; $\mathbf{P}$ and $\mathbf{Q}$ are the loading vectors of input matrix $\mathbf{X}$ and output matrix $\mathbf{Y}$, respectively; m and n are the number of process variables and observations in $\mathbf{X}$; p is the number of quality variables in $\mathbf{Y}$ and M the total number of extracted latent variables; $\boldsymbol{s}$ and $\boldsymbol{u}$ are the input and output score vectors; and $\boldsymbol{p}$ and $\boldsymbol{q}$ are the input and output loading vectors.

First, the input and output matrices are standardized. To initialize the algorithm, one column of the output data matrix is assigned as output score vector. At each iteration $\boldsymbol{s}$, $\boldsymbol{u}$, $\boldsymbol{p}$ and $\boldsymbol{q}$ are calculated and stored while M latent variables are extracted. Iteration is continued until convergence is reached. (Ibid)

The main limitations of both standard PCA and PLS techniques is their inability to cope with nonlinear and dynamic processes. Numerous variants of both techniques suitable for nonlinear and dynamic problems can be found in the literature, such as dynamic PCA and PLS, dynamic nonlinear PCA and PLS, dynamic kernel PCA, and multiblock PCA (Das et al. 2012, 728; Tidriri et al. 2016, 65–66). PCA and PLS techniques are used together with control charts to assign appropriate threshold limits for alarms and detecting faults. The two most widely used fault detection indices are the T^2 and Q statistics. The T^2 chart represents the captured variation within the model plane indicating the distance of the current operation from the desired operation. The desired state of operation is described by the model and the corresponding principal components or latent variables. The Q statistics, also known as squared prediction error (SPE), represents the variance not captured by the model. These

two charts must be used as a pair, as a significant deviation captured by either of the charts indicates abnormal operation. The main advantage of using PCA and PLS together with the control charts is that instead of monitoring many parameters at once, only the evaluation of the two chart output variables is required to process monitoring (Yang 2004, 62). (MacGregor & Cinar, 2012, 112–113)

For fault diagnosis, PCA and PLS can also be used with the T^2 and Q statistics, along with their variants (Qiu & Bai, 2012). This approach usually has one PCA model for each system behavior, i.e. for normal operation and for each faulty situation, which enables classification of the model outputs (Villegas et al. 2010; Yang 2004, 67–69). Another practice for fault diagnosis is to use the outputs of PCA and PLS models as inputs for other classification tools, such as hidden Markov models (Zhou et al. 2004) or neural networks (Zhang et al. 2013). Other standard methods for diagnosis are contribution plots, which show the contribution of each process variable to the calculated statistic (Westerhuis & et al. 2000, 95), and reconstruction-based contributions (Fuentes-Garcia et al. 2018, 196).

5.2.2 Artificial neural networks

Artificial neural networks (ANN) are a widely used method, also in condition monitoring, inspired by the human brain structure. They can represent highly nonlinear functions and perform multi-input, multi-output mapping, making them very powerful in classification and pattern recognition (Van Tung & Yang, 2009, 63). ANNs have two important features that make them very valuable in many applications. First, ANNs can approximate any nonlinear function with arbitrary accuracy with suitable architecture and weight parameters. Second, they are capable of self-learning by extracting features from training data without a priori knowledge about the studied process. (Patan 2008, 15)

The computational model of an ANN consists of processing elements called neurons, and connections with weight parameter between neurons called links. Each neuron in the network receives stimulus from the neighboring connected neurons, processes the stimulus and creates output. A schematic ANN structure is illustrated in Figure 16, where the neurons are arranged in different layers, namely input layer, one or multiple hidden layers and output layer. The network can be structured, and the neurons can process information in various

ways, making ANN methods very versatile. (Ismail et al. 2016, 1; Zhang & Gupta, 2000, 61)

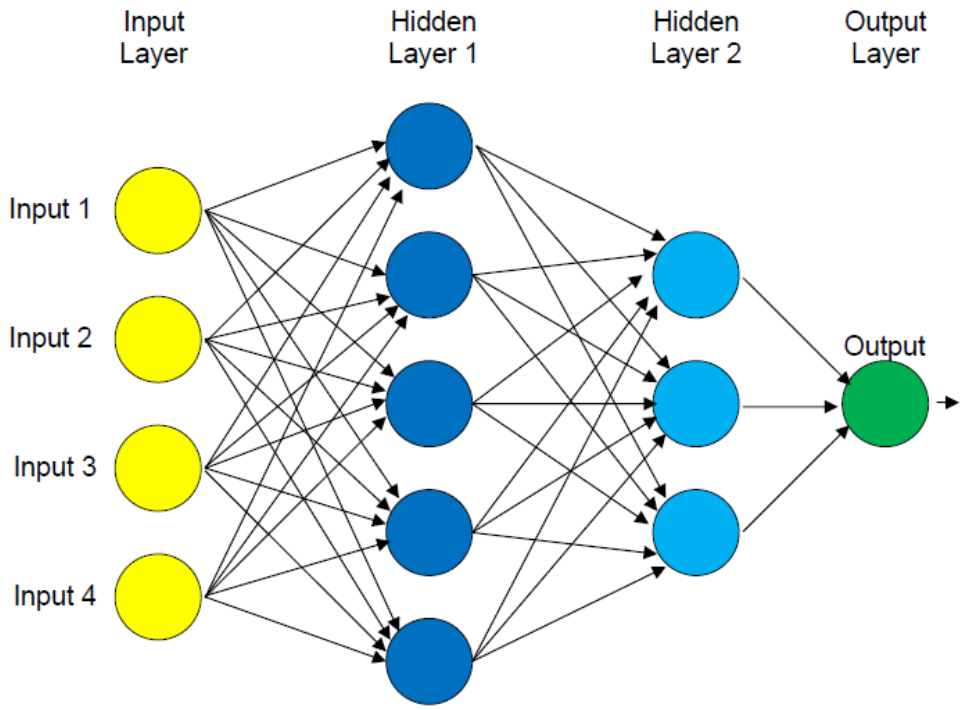


Figure 16. Simple feedforward ANN structure with two hidden layers (Ismail et al. 2016, 2).

ANN is trained with a training data set. During this process, the ANN learns the unknown function and optimizes the network parameters by adjusting the weights according to input and output observations (Van Tung & Yang, 2009, 63; Sutharssan et al. 2015, 5). The training can be supervised or unsupervised. In supervised learning, external input is required about the desired output. For instance, common practice is to use data sets with known faults to train an ANN to detect and diagnose these fault symptoms in actual operation. In unsupervised learning such a priori knowledge is not required but the ANN learns itself using the new available information. Another way to classify ANNs is based on the network architecture, such as multilayer perceptrons (MLP), self-organizing maps (SOM) and radial basis function networks, among others. The main limitations of any ANN are mostly due to weak selection of network weights and parameters or the number of hidden neurons (Talebi et al. 2010, 8). (Khoukhi & Khalid, 2015, 19; Zhang & Gupta, 2000, 62)

MLP with back propagation training algorithm is one of the most used ANN approaches. MLP has a feed-forward structure, as illustrated in Figure 16. Runtime data is fed to the input neurons where from it propagates via connections to the hidden neurons in one or multiple hidden layers and finally to the output neurons. Each neuron takes a weighted sum of its inputs and the sum is fed to the neuron activation function to compute the final output of the neuron, which can be the input of neurons in the following layer. The neuron activation function can have several forms, such as linear thresholds and Gaussian functions, sigmoidal functions, see Figure 17, being the most used one. (Van Tung & Yang, 2009, 63; Stanley 2013; Talebi et al. 2010, 7–11)

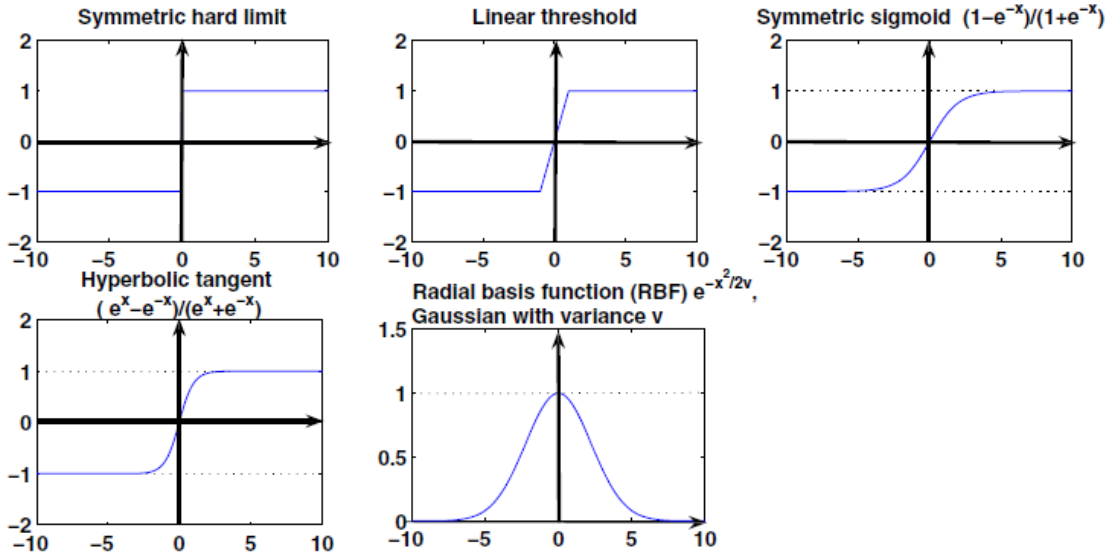


Figure 17. Common activation functions of MLP networks (Talebi et al. 2010, 9)

The nonlinear computing of an MLP illustrated in Figure 18 can be expressed as

$$\boldsymbol{y} = \sigma_3\{\mathbf{W}_3\sigma_2[\mathbf{W}_2\sigma_1(\mathbf{W}_1\boldsymbol{u})]\}, \tag{7}$$

where σ_1, σ_2 and σ_3 are vector valued activation functions; $\mathbf{W}_1$, $\mathbf{W}_2$ and $\mathbf{W}_3$ are the weight coefficient matrices; $\boldsymbol{u}$ is the input vector and $\boldsymbol{y}$ the output vector (Patan 2008, 17).

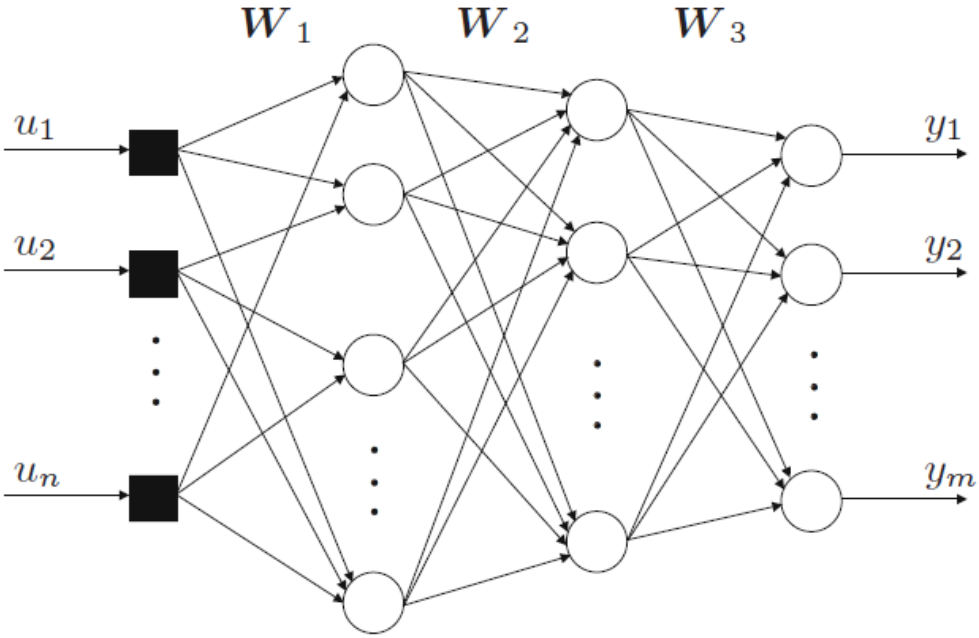


Figure 18. A three-layer perceptron with n inputs and m outputs (Patan 2008, 18).

Training of an ANN is basically a process to determinate the weight coefficients that define connection intensity between neurons. The fundamental back propagation training algorithm is a form of supervised learning. The iterative algorithm uses the optimization gradient descent method to minimize a sum-squared error. The weights inside the network are adjusted based on the equation

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \alpha \nabla J\big(\boldsymbol{w}(k)\big), \tag{8}$$

where $\boldsymbol{w}(k)$ denotes the weight vector at discrete time k, α is the user-defined learning rate and $\nabla J(\boldsymbol{w}(k))$ is the gradient of the performance index J with respect to the weight vector $\boldsymbol{w}$. In the training phase, the error of the output layer is propagated towards the input layer and the weights inside the network are adjusted based on the gradient of the performance index. (Patan 2008, 18)

It is worthwhile to distinguish between shallow (one hidden layer) and deep (two or multiple hidden layers) networks. While it is proven that a multilayer feedforward network with only one hidden layer can approximate any nonlinear function (Hornik 1990) this can still be very difficult or impossible in practice, since the required number of hidden neurons can grow

unfeasibly large. Deep neural networks can reduce the required number of hidden neurons and they have become increasingly popular in recent years and proven very efficient in many fields. (Goodfellow et al. 2016, 195–197).

Feedforward networks can functionally map input and output data points, but the standard network is static (Malhi & Gao, 2004, 2048). Most of industrial processes are dynamic and to take this into account, the ANN requires either external memory or feedback. The most used network type with external memory is a multilayer perceptron type time-delay network, which has two parts: a non-linear static approximator and an external dynamic filter bank. These types of networks suffer from stability issues and are not as popular as recurrent networks with feedback loops. Recurrent neural networks are specialized for processing sequential data, which makes them convenient for time-series prediction, such as RUL prediction. Recurrent networks can store information from previous time-steps by having feedback connections from the hidden or output layers to the previous layers (Malhi & Gao, 2004, 2049). This network structure, illustrated in Figure 19, enables neuron to receive external inputs and also state feedback signals from itself and other neurons in the network. Based on the location of feedback loops, recursive networks can be divided into globally (feedback on global network level) and locally (feedback on neuron level) recurrent networks. A variation of the standard back-propagation training algorithm, called back-propagation through time is usually applied to recurrent ANNs. (Patan 2008, 29–36; Goodfellow et al. 2016, 367–379)

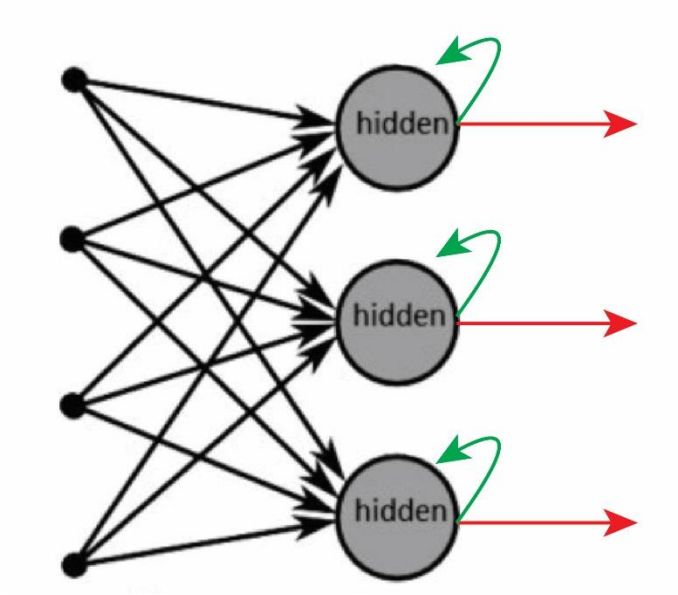


Figure 19. A locally recurrent neural network. The hidden neurons receive external inputs (black arrows) and feedback loops from previous time-steps (green arrows) creating output (red arrows). Adopted from Radhakrishnan (2017).

For fault detection, ANNs can be used similarly to other methods by comparing measured values to expected values. Usually the network is trained with healthy state data, and after the training the network is run parallel to the monitored system to detect abnormalities in the form of residuals. The residuals can also be used as input data for a second network for fault isolation. The other used approach is based on a clear differentiation of various faults from healthy state operation also including the differentiation between various faults. This approach uses supervised learning algorithms to identify a priori defined fault patterns in actual operation. (Hush et al. 1997, 918–919; Das et al. 2012 ,729)

Fault diagnosis with ANNs is basically a classification problem. Instead of measurements or process characteristics, fault patterns are used as network inputs producing classes or fault categories. For example, the output classes can be labeled as healthy, anomaly and so on. Regression can then be used to extrapolate the damage or fault propagation to estimate the RUL of the monitored system. (Das et al. 2012 ,729; Sutharssan et al. 2015, 5)

5.3 Hybrid modelling

The above discussed modeling methods to condition monitoring each have their own advantages and disadvantages summarized in Table 2. Many different methods have been used in conjunction to overcome these limitations. Combining different modeling methods is known as hybrid modeling. Hybrid models are considered a promising approach, as different models complement each other. Hybrid models can be constructed in various ways. For example, neuro-fuzzy systems combining ANNs and fuzzy logic systems have been widely used. This approach utilizes the learning capability of ANNs and the ability of fuzzy logic systems to represent human knowledge (Van Tung & Yang, 2009, 62). Another approach is to use physics-based models for fault detection combined with data-driven fault diagnosis (Das et al. 2012, 737). Physics-based models can also be used to generate fault signals, which can be used as training data for machine learning approaches. Such an approach was proposed, for example, by Garcia-Matos et al. (2013), as they used a physical model to create a fault dictionary to be compared with faults detected using an MLP model to monitor an axial compressor. Other examples of hybrid models for power plants are proposed by Simani & Fantucci (2000) and Chen et al. (2015). Simani & Fantucci proposed Kalman filters to residual generation and fault detection integrated with fault identification with ANNs and used a physics-based power plant model for model testing. Chen et al. integrated generalized regression neural network (GRNN), mean impact value (MIV), partial least squares regression (PLSR) and B-spline transformation techniques to monitor sensors in power plants. The GRNN model is used to assess the average contribution rate of independent variables and the MIV method is used to filter out the main modeling parameters. These parameters are then modeled using the PLSR method based on the B-spline transformation.

Table 2. Summary of advantages and disadvantages of different CM approaches.

Approach	Advantages	Disadvantages
Physics-based models	can be highly accurate explains process behavior can be extrapolated to different operation conditions requires less data compared to empirical models	complex to create and solve simplifying assumptions reduce accuracy requires determination of various physics parameters
Statistical models	no a prior process knowledge is needed easy to develop and implement can provide confidence intervals of new observations resistant to input noise	often requires assumptions of the underlying statistical characteristics cannot explain process behavior requires large amount of data cannot be extrapolated to different operation conditions
Artificial neural networks	no a prior process knowledge is needed can model complex nonlinear processes short development time resistant to input noise	cannot explain process behavior applicable only to operation mode used for training cannot detect and diagnose noble faults requires large amount of data training process is prone to errors

6 CASE STUDY: DETECTION OF ABNORMALITIES WITH ARTIFICIAL NEURAL NETWORK BASED SOFTWARE

In this section, a commercial condition monitoring software Intelligent Health Monitor (IHM), introduced in Section 6.1, is evaluated in practice in a case study with process data from a reference power plant. The owner and location of the reference plant are confidential.

The gas-powered combined heat and power plant with gross thermal output of approximately 250 MW was commissioned in the 2000s.

6.1 Intelligent Health Monitor

Intelligent Health Monitor is a commercial condition monitoring software by algorithmica technologies Gmbh (algorithmica technologies Gmbh 2018). IHM applies machine learning in the form of recurrent neural networks to create a mathematical model for each parameter, called tags, in terms of the others. The monitored parameters do not need to be actual measurements. Tags can be any parameters included in the database, for example calculated process parameters such as heat transfer coefficients derived from measurements. After the model is created, dynamic alarm limits are employed to the monitored parameter. The dynamic alarm limits are computed from expected values and the chosen confidence interval of the model. (Bangert 2018)

IHM is based on machine learning and requires training data from healthy state operation. Ideally, the training data set would include all relevant operation conditions, i.e. load variations and startups, for the model to be accurate monitoring the actual operation. The independent variables used to model each monitored parameter can be chosen manually by the user or automatically based on a combination of correlation modeling and principal component analysis. Once the model is created, it can be verified using second data set not included in the training of the network. The model detects deviations from the expected healthy behavior and can be used in online condition monitoring or in offline analysis. (Ibid)

6.2 Case study: Decreased boiler efficiency

Boiler thermal efficiency is one of the key parameters to monitor in a thermal power plant. The efficiency directly affects to the operating costs of the plant as boilers operating with low efficiency require more fuel to generate the desired heat output. The boiler efficiency can be calculated two ways. The direct way is to calculate the efficiency as the ratio of the useful heat absorbed into the steam Q_{abs} and the heat and energy brought into the boiler Q_{in} as:

$$\eta = \frac{Q_{\text{abs}}}{Q_{\text{in}}} \tag{9}$$

Another option is to calculate the efficiency indirectly as:

$$\eta = 1 - \sum Q_{\text{loss},i} \tag{10}$$

where $Q_{\text{loss},i}$ denotes individual heat losses. The hot flue gas out of the boiler represents the largest individual heat loss, which is proportional to the mass flow and the temperature of the flue gas. (Huhtinen et al. 1994, 92–99) In this case study, the flue gas temperature after the final superheater before the economizer is monitored. The flue gas temperature before the economizer has gradually increased, which indicates decreased performance of the boiler heat surfaces prior to the economizer.

6.2.1 Problem description

Historical data of the flue gas temperature before economizer is monitored using 96 different models. The used models are created using different parameters in the model creation. The varied parameters are the training data period, the number of hidden layers and neurons in the ANN architecture and the independent (input) variables used to model the monitored parameter. The training parameters are shown in Table 3 and in Table 4. The used training data periods are one month, four months and twelve months. The models are created using two or four (software's default setting) hidden layers. Number of hidden neurons in the first hidden layer are chosen based on the number of independent variables. The chosen number of hidden neurons in the first hidden layer are 0.8, 1.5, 2, and 3 times the number of independent variables. The number of hidden neurons is halved in each consecutive layer. For example, in the case of 30 hidden neurons in the first hidden layer, the second layer has 15 hidden neurons, the third layer has 8 hidden neurons, and the fourth layer has 4 hidden neurons. Four different ways to choose the independent variables were chosen. 10 and 30 independent variables are chosen automatically by the software based on PCA and correlation analysis. In addition, 10 and 30 independent variables are chosen manually based on process knowledge. The independent variables in each case are shown in Table 5. Some of the variables in Table 5 have parallel measurements, e.g. several redundant temperature measurements at the same location. Also, similar components, such as superheaters, are mentioned only once in the list, although several of them are included in the models.

Table 3. Training parameters for models with 2 hidden layers.

Training period: 1 month			Training period: 4 months			Training period: 12 months		
Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer	Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer	Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer
1AA	10 Auto	8	1BA	10 Auto	8	1CA	10 Auto	8
1AB	10 Auto	15	1BB	10 Auto	15	1CB	10 Auto	15
1AC	10 Auto	20	1BC	10 Auto	20	1CC	10 Auto	20
1AD	10 Auto	30	1BD	10 Auto	30	1CD	10 Auto	30
1AE	30 Auto	24	1BE	30 Auto	24	1CE	30 Auto	24
1AF	30 Auto	45	1BF	30 Auto	45	1CF	30 Auto	45
1AG	30 Auto	60	1BG	30 Auto	60	1CG	30 Auto	60
1AH	30 Auto	90	1BH	30 Auto	90	1CH	30 Auto	90
1AI	10 Man	8	1BI	10 Man	8	1CI	10 Man	8
1AJ	10 Man	15	1BJ	10 Man	15	1CJ	10 Man	15
1AK	10 Man	20	1BK	10 Man	20	1CK	10 Man	20
1AL	10 Man	30	1BL	10 Man	30	1CL	10 Man	30
1AM	30 Man	24	1BM	30 Man	24	1CM	30 Man	24
1AN	30 Man	45	1BN	30 Man	45	1CN	30 Man	45
1AO	30 Man	60	1BO	30 Man	60	1CO	30 Man	60
1AP	30 Man	90	1BP	30 Man	90	1CP	30 Man	90

Table 4. Training parameters for models with 4 hidden layers.

Training period: 1 month			Training period: 4 months			Training period: 12 months		
Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer	Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer	Model	Independ-ent varia-bles	Number of hidden neurons in the first hidden layer
2AA	10 Auto	8	2BA	10 Auto	8	2CA	10 Auto	8
2AB	10 Auto	15	2BB	10 Auto	15	2CB	10 Auto	15
2AC	10 Auto	20	2BC	10 Auto	20	2CC	10 Auto	20
2AD	10 Auto	30	2BD	10 Auto	30	2CD	10 Auto	30
2AE	30 Auto	24	2BE	30 Auto	24	2CE	30 Auto	24
2AF	30 Auto	45	2BF	30 Auto	45	2CF	30 Auto	45
2AG	30 Auto	60	2BG	30 Auto	60	2CG	30 Auto	60
2AH	30 Auto	90	2BH	30 Auto	90	2CH	30 Auto	90
2AI	10 Man	8	2BI	10 Man	8	2CI	10 Man	8
2AJ	10 Man	15	2BJ	10 Man	15	2CJ	10 Man	15
2AK	10 Man	20	2BK	10 Man	20	2CK	10 Man	20
2AL	10 Man	30	2BL	10 Man	30	2CL	10 Man	30
2AM	30 Man	24	2BM	30 Man	24	2CM	30 Man	24
2AN	30 Man	45	2BN	30 Man	45	2CN	30 Man	45
2AO	30 Man	60	2BO	30 Man	60	2CO	30 Man	60
2AP	30 Man	90	2BP	30 Man	90	2CP	30 Man	90

Table 5. Independent variables in model creation. Numbers inside parentheses () refer to the total number of measurements corresponding to similar components and/or parallel measurements.

10 Automatic	30 Automatic	10 Manual	30 Manual
Flue gas volumetric flow Boiler steam power (2) Combustion air volumetric flow Feed water mass flow Main steam mass flow Fuel mass flow (2) Fuel power Boiler selection switch	Flue gas volumetric flow (2) Boiler steam power (2) Combustion air volumetric flow Feed water mass flow (2) Main steam mass flow (2) Fuel mass flow (3) Fuel power Boiler selection switch Flue gas mass flow Combustion air mass flow Combustion air fan current Spray water mass flow Steam temperature after superheater Fuel gas strainer pressure difference NO2 mass emissions (2) Furnace pressure (3) Flue gas temperature to stack Combustion air pressure Emission measurement temperature Superheater material temperature Drum level	Main steam pressure Main steam temperature Superheater material temperature (3) Steam temperature after superheater (5)	Main steam pressure Main steam temperature Superheater material temperature (9) Steam temperature after superheater (6) Drum pressure (2) Feed water pressure Feed water temperature after economizer Drum material temperature (3) Feed water temperature Steam temperature superheater drain (3) Flue gas temperature to stack (2)

6.2.2 Evaluation criteria

Each created model is evaluated based on graphical inspection of the model trend. The models are evaluated using the following hierarchical criteria:

1. Model fit to training data
2. Model fit to monitored healthy data
3. Abnormality detection

The model needs to meet all the above criteria to be considered reliable and accurate. The evaluation criteria is hierarchical in a sense that each model has to meet the previous criteria before it is reasonable to evaluate the next criterion. I.e., a model capable of detecting the increased flue gas temperature is still useless in practice, if the model is not capable of recognizing healthy behavior during the training period or in monitoring phase.

Each of the three criteria are evaluated on the following scale:

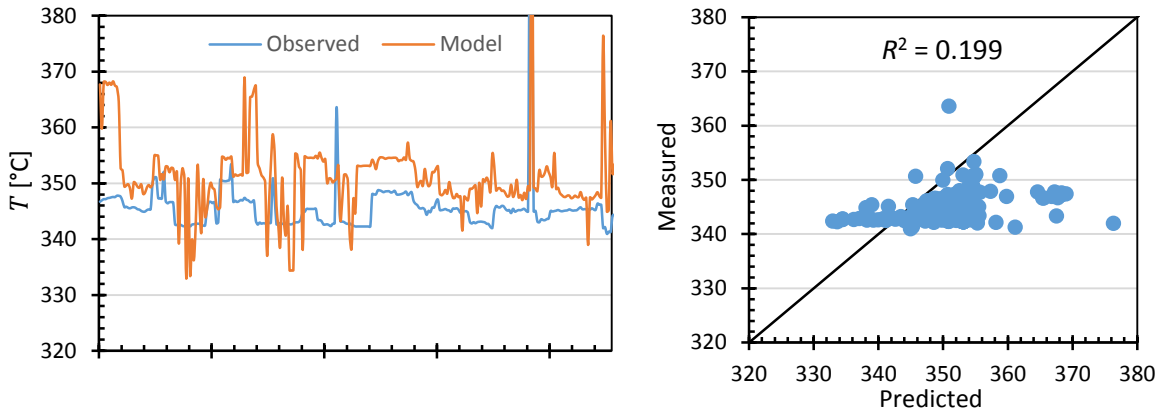
1. Very poor
2. Poor
3. Fair
4. Good
5. Very Good

During the training period and healthy state monitoring phase the models are evaluated as follows. Models with constant very significant deviation (over 10 Celsius degrees) from measured values or with clearly inconsistent prediction are treated as very poor. Poor models follow the measured trend to some extent but are characterized by constant significant deviations (from 5 to 10 Celsius degrees) and occasional very significant deviations from the measured values. Models with decent overall production are treated as fair. These models can capture the process dynamics but are characterized by constant minor deviations (from 3 to 5 Celsius degrees) and occasional significant or very significant deviations. Models that most of the time have accurate prediction with often occurring minor deviations are treated as good. Very good models follow the measured trend very well and have only few minor deviations from measured values. During the last phase, i.e. abnormality detection, the above criteria is used. In addition, the model's performance is judged based on the level of the

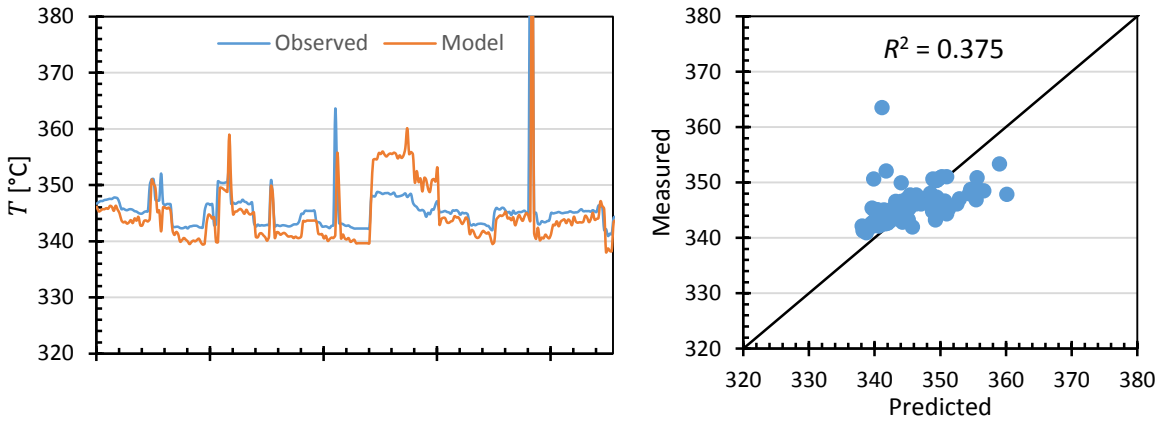
predicted trend. Grade “good” requires trend prediction with similar behavior as the measured values with an indication of the increased temperature. The model evaluation criteria for each phase are summarized in Table 6 and examples of each fit are illustrated in Figure 20.

Table 6. Model evaluation criteria.

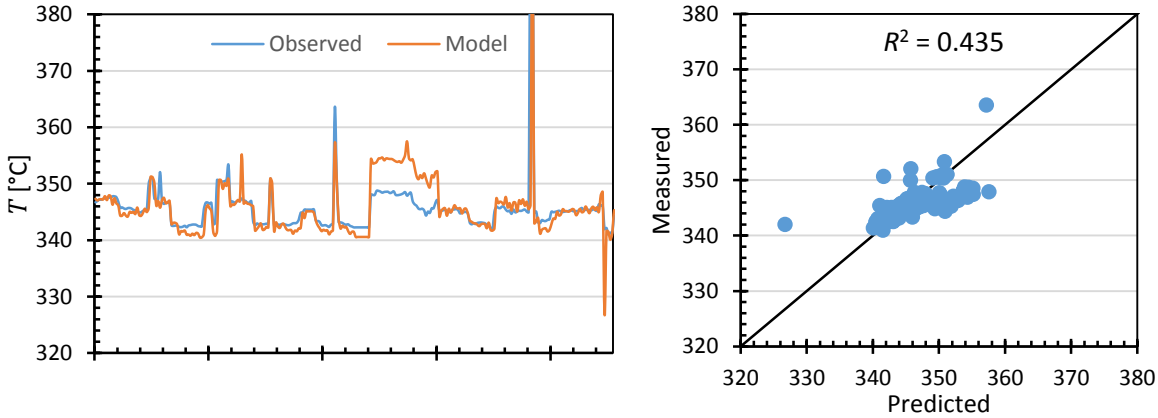
	Fit to training/monitored data	Abnormality detection
Very poor	Constant inconsistent prediction Constant very significant deviation	Cannot be detected, model is not reliable
Poor	Constant significant deviation Occasional very significant deviation	Cannot be detected, model follows measured trend at the same or at higher level
Fair	Constant minor deviation Occasional significant or very significant deviation	Minor indication, model follows measured trend
Good	Often occurring minor deviation	Indication that results in a more detailed analysis
Very good	Occasional minor deviations	Clear indication



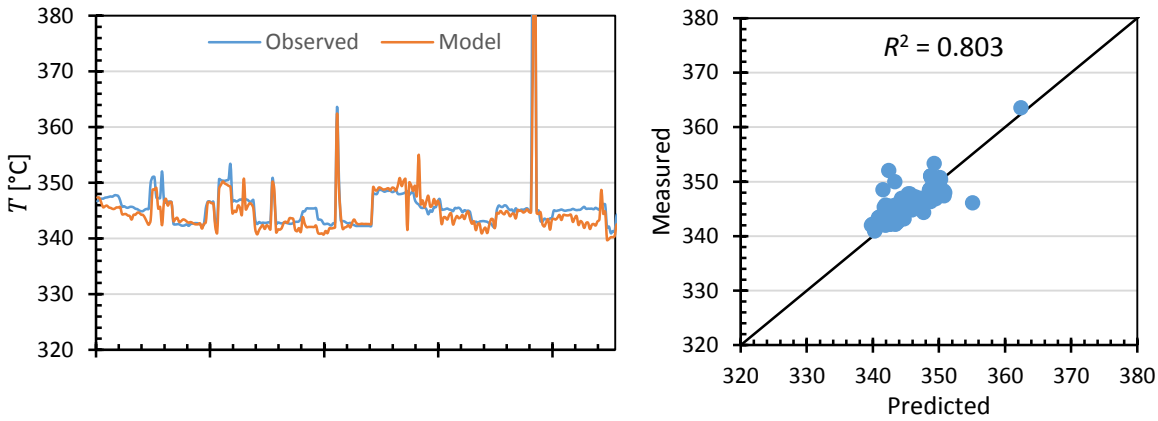
a) Very poor fit (Model 1AN)



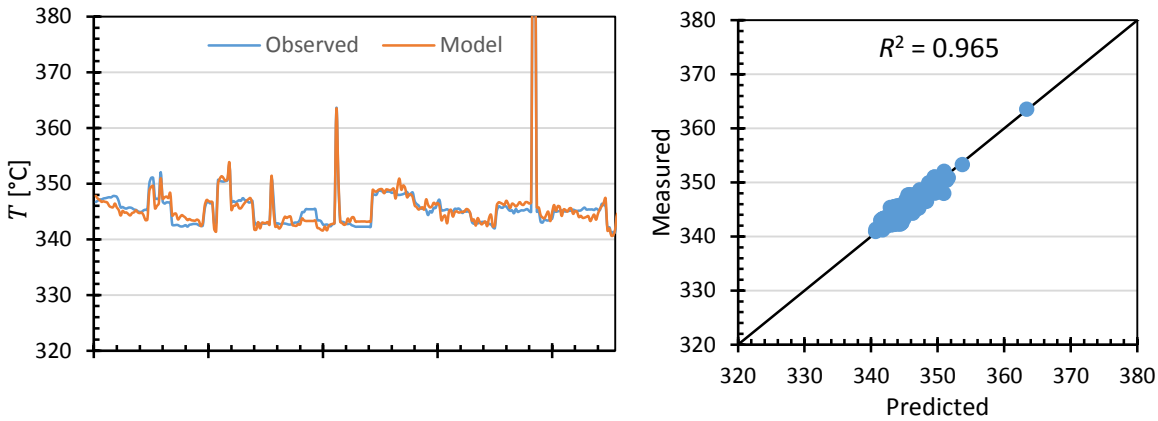
a) Poor fit (Model 1CC)



b) Fair fit (Model 1CD)



c) Good fit (Model 1CE)



d) Very good fit (Model 2CH)

Figure 20. Examples of the scale used to evaluate the model fit. Each plot corresponds to a period of three months during the monitoring phase.

6.2.3 Results

The models need to have the grade of "good" or "very good" in each evaluated category to be considered useful and accurate enough to be used in actual plant monitoring. 13 of the 96 models met this criterion. Model 2CH with training period of 12 months, 30 automatically chosen independent variables, 4 hidden layers, and 90 hidden neurons showed very good performance and was chosen as a reference model, see Figures 21–24. The evaluation results for models with 2 hidden layers are shown in Table 7 and for models with 4 hidden layers in Table 8.

Figure 21. Measured and predicted flue gas temperature before economizer with reference model 2CH over 35 months. The dashed lines mark the three periods: training, healthy state operation and abnormality detection.

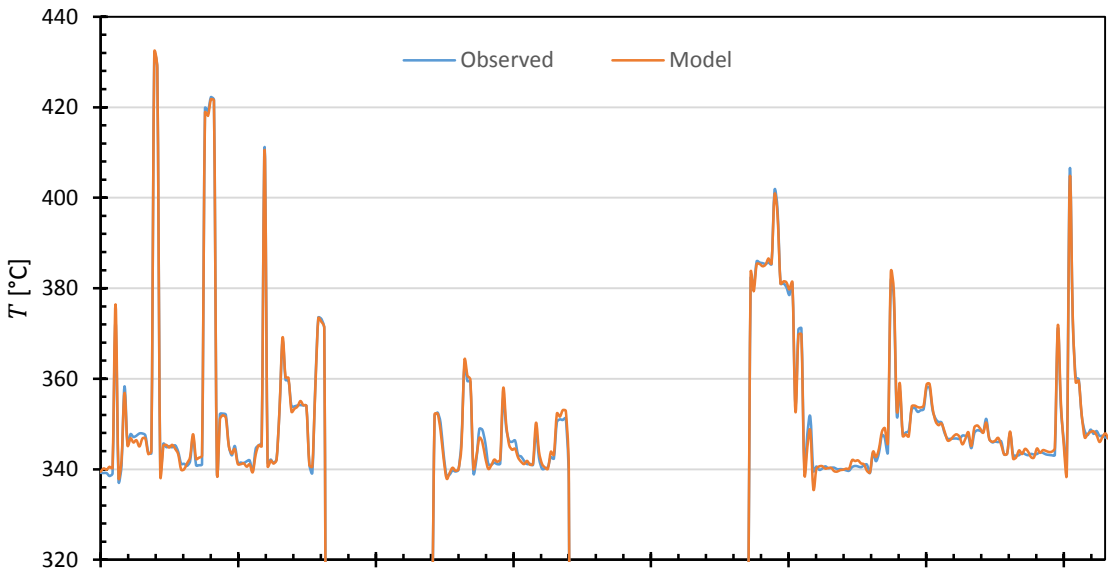


Figure 22. Measured and predicted flue gas temperature before economizer with reference model 2CH during the training period of 12 months.

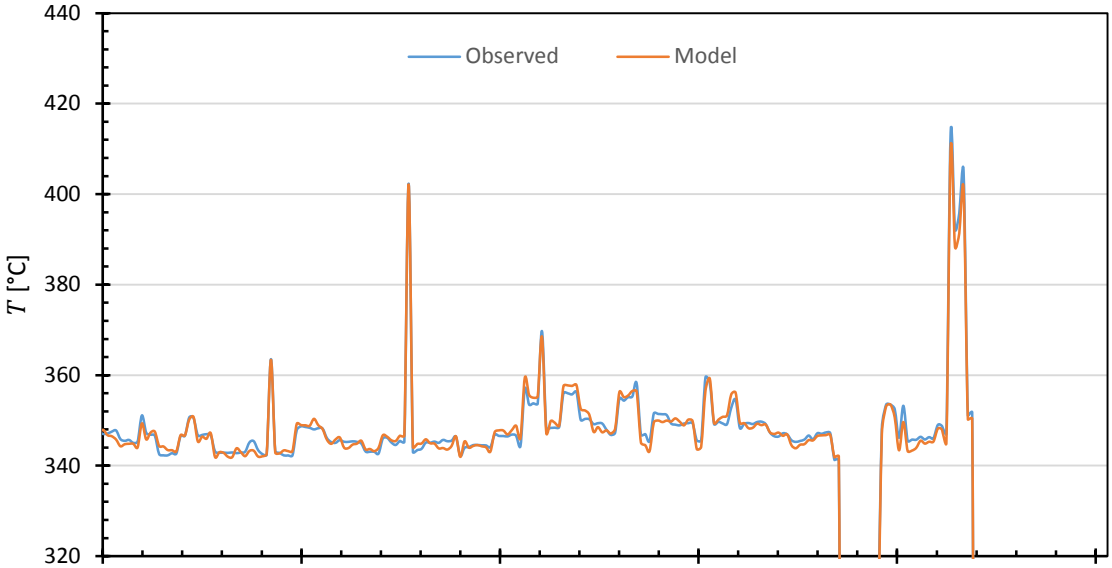


Figure 23. Measured and predicted flue gas temperature before economizer with the reference model 2CH during the monitoring phase of 10 months (assumed healthy state).

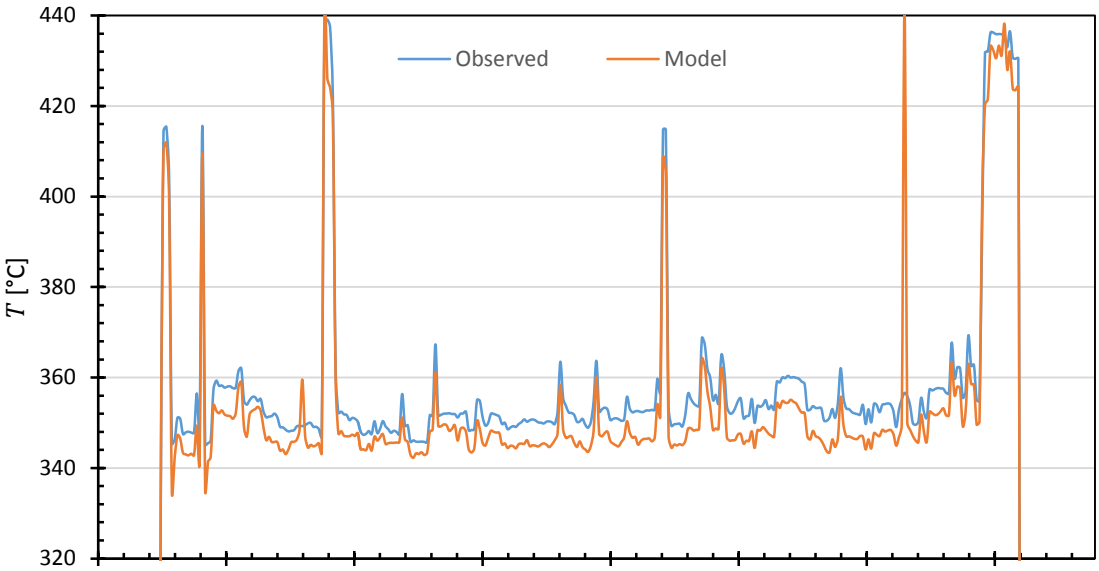


Figure 24. Measured and predicted flue gas temperature before economizer with reference model 2CH during abnormal operation of 13 months.

Table 7. Case study evaluation results of models with 2 hidden layers. Models with good overall performance are highlighted.

Training period: 1 month				Training period: 4 months				Training period: 12 months			
Model	Fit to training data	Fit to monitored data	Abnormality detection	Model	Fit to training data	Fit to monitored data	Abnormality detection	Model	Fit to training data	Fit to monitored data	Abnormality detection
1AA	Very good	Fair	Poor	1BA	Good	Very good	Poor	1CA	Fair	Fair	Poor
1AB	Very good	Fair	Poor	1BB	Good	Very good	Poor	1CB	Fair	Fair	Poor
1AC	Very good	Fair	Poor	1BC	Very good	Very good	Poor	1CC	Fair	Fair	Poor
1AD	Very good	Fair	Fair	1BD	Good	Very good	Poor	1CD	Poor	Fair	Poor
1AE	Very good	Fair	Poor	1BE	Very good	Very good	Very good	1CE	Fair	Fair	Very good
1AF	Very good	Poor	Poor	1BF	Very good	Very good	Very good	1CF	Fair	Fair	Poor
1AG	Very good	Fair	Poor	1BG	Very good	Very good	Very good	1CG	Fair	Fair	Very good
1AH	Very good	Fair	Poor	1BH	Very good	Very good	Fair	1CH	Fair	Good	Poor
1AI	Very good	Very poor	Very good	1BI	Fair	Fair	Poor	1CI	Poor	Poor	Very poor
1AJ	Very good	Poor	Good	1BJ	Good	Fair	Poor	1CJ	Poor	Poor	Very poor
1AK	Very good	Very poor	Very poor	1BK	Very good	Fair	Poor	1CK	Poor	Poor	Poor
1AL	Very good	Very poor	Very poor	1BL	Very good	Fair	Very poor	1CL	Poor	Fair	Very poor
1AM	Very good	Very poor	Very poor	1BM	Very good	Poor	Fair	1CM	Poor	Fair	Poor
1AN	Very good	Very poor	Very poor	1BN	Very good	Fair	Fair	1CN	Fair	Poor	Poor
1AO	Very good	Very poor	Very poor	1BO	Very good	Fair	Very poor	1CO	Fair	Fair	Fair
1AP	Very good	Very poor	Very poor	1BP	Very good	Poor	Very poor	1CP	Good	Fair	Fair

Table 8. Case study evaluation results of models with 4 hidden layers. Models with good overall performance are highlighted.

Training period: 1 month				Training period: 4 months				Training period: 12 months			
Model	Fit to training data	Fit to monitored data	Abnormality detection	Model	Fit to training data	Fit to monitored data	Abnormality detection	Model	Fit to training data	Fit to monitored data	Abnormality detection
2AA	Fair	Good	Fair	2BA	Very good	Very good	Poor	2CA	Very poor	Very poor	Very poor
2AB	Fair	Poor	Poor	2BB	Very poor	Very poor	Very poor	2CB	Very poor	Very poor	Very poor
2AC	Very good	Poor	Fair	2BC	Very good	Good	Fair	2CC	Fair	Poor	Poor
2AD	Very good	Poor	Poor	2BD	Very good	Good	Poor	2CD	Poor	Fair	Poor
2AE	Good	Fair	Poor	2BE	Very good	Very good	Very good	2CE	Good	Good	Poor
2AF	Very good	Poor	Poor	2BF	Very good	Good	Good	2CF	Very good	Very good	Good
2AG	Very good	Fair	Poor	2BG	Very good	Very good	Good	2CG	Very good	Very good	Very good
2AH	Very good	Fair	Poor	2BH	Very good	Very good	Very good	2CH	Very good	Very good	Very good
2AI	Fair	Poor	Poor	2BI	Very poor	Very poor	Poor	2CI	Very poor	Very poor	Very poor
2AJ	Very good	Poor	Very good	2BJ	Very good	Fair	Very poor	2CJ	Poor	Poor	Fair
2AK	Fair	Very poor	Very good	2BK	Very good	Very good	Poor	2CK	Poor	Good	Poor
2AL	Very good	Poor	Very good	2BL	Very good	Fair	Poor	2CL	Poor	Poor	Poor
2AM	Very good	Very good	Very good	2BM	Very good	Poor	Very good	2CM	Fair	Fair	Fair
2AN	Very good	Very poor	Very good	2BN	Very good	Good	Poor	2CN	Fair	Fair	Fair
2AO	Very good	Very poor	Very good	2BO	Very good	Fair	Fair	2CO	Very good	Very good	Very good
2AP	Very good	Very poor	Very good	2BP	Very good	Poor	Very poor	2CP	Good	Good	Very good

Models with a training period of one month show similar behavior regardless of other training parameters. These models are characterized by good fit to the training data but inaccurate prediction in actual monitoring. In this case, the training period of one month is too short for the network to learn the process behavior. As a result from the small training data and complex network structure, these models most likely suffer from overfitting. I.e., the models have “memorized” the process behavior rather than learned the underlying statistics. Model 2AM (4 hidden layers, 30 independent variables and 24 hidden neurons in the first hidden layer) is the only exception with very good performance.

Models with a training period of four months show better prediction capability. These models also fit very well to the training data while their performance on additional data sets varies. Models 1BE–1BG and 2BE–2BH with 30 hidden neurons in the first hidden layer have very good overall performance. These models form the best performing group as they account for 7/13 of the reliable models.

The performance of models with a training period of 12 months varies the most. Models with two hidden layers fail to predict the flue gas temperature during all three phases. The model accuracy clearly increases with the added two hidden layers. There is still a lot of variation among these models, as some models are reliable while the majority fails to learn the temperature behavior.

In general, the model performance significantly varies, no matter how the models are classified. However, some conclusions can be made based on the results. First, models with 10 automatically or manually chosen independent variables are all inaccurate. In this case, with the chosen architectures, the selected 10 input variables are not enough to capture the process dynamics. Second, the 30 automatically chosen input variables seem to be most suitable of the chosen options, even though these variables include some variables without any physical relation to the monitored parameter. Third, models with a training period of one month have the worst accuracy but increasing the training period from four to twelve months does not seem to increase the models’ predictive power. The impact of the number of hidden neurons on the model performance considerably varies and no conclusions on this relation can be made.

7 ARTIFICIAL NEURAL NETWORK BASED CONDITION MONITORING

The implementation of ANN based monitoring into a condition monitoring system can be divided into three parts: training phase, verification phase and monitoring phase. This section discusses the practical aspects that need to be considered in the training and model verification phases. The monitoring phase is relatively straight-forward from technical point of view and is not discussed in this section. The considered aspects are illustrated with the case study from Section 6.

7.1 ANN training

The performance and accuracy of any ANN is determined by the quality of the network training. ANNs are often described as "black-box" models, since the network's data processing is complex and often difficult to comprehend (May et al. 21, 2011). Commercial software can strengthen this image as the user has limited possibilities to vary the training parameters or the network structure and to investigate the created models. Nevertheless, it is vital for the user to have basic understanding on how different parameters affect on the network performance. It is also important to distinguish between different ANN types and specific engineering problems. This section considers ANN's used in regression problems and more specifically time-series prediction.

The training parameters of ANN's are highly problem specific and no universal solution for optimal parameter selection exists (Matignon 2005, 147). Desired training parameters result in a model that is computationally efficient and generalizes well. Generalization is the most desired network property, which means that the network can identify the dependencies of the training data in new data sets introduced in the monitoring phase. Insufficient training parameters may result in overtraining or overfitting. Overtraining refers to a situation, where the network training is continued beyond optimal point (Tzafestas et al. 1996, 508) and overfitting to exceeding some optimal ANN size. In both situations, the network "memorizes" the patterns, including noise, present in the training data set and loses the prediction capability (Tetko et al. 1995, 827). Overfitting results from using too complex network structure with excessive degrees of freedom (Alman & Ningfang, 2001, 121). This problem stems from the so-called curse of dimensionality meaning that if the dimension of input space is

high, an exponentially large number of observations is required for optimal generalization and the network uses most of its recourses to represent irrelevant portions of the input space (Bach, 2014, 2). It is therefore important to adjust the training parameters accordingly to the monitored system. Both over-simplified and too complex models are to be avoided.

7.1.1 Data requirements and training period

All machine learning algorithms, including ANNs, learn the system behavior from training data, which makes the selection of appropriate training data a vital part of the training process. The first consideration is to determine what kind of operation conditions are monitored. These same conditions, e.g. load and environmental conditions, should be present in the training data. The second important consideration is to determine healthy state of the monitored parameter. If the training data includes faulty behavior of the monitored equipment, the model can learn the behavior and won't recognize it as abnormal in the monitoring phase. Careful inspection of the training data is therefore required to avoid training unwanted process behavior to the model. This task can be challenging, especially for slowly developing faults. The longer the available historical time period, the easier this task is.

The determination of healthy state is also closely related to the question of when and how often the model should be updated. Most commercial condition monitoring software based on machine learning can be categorized into continuously updated models and periodically updated models. Software in the first category continuously adjust the network based on constant acquisition of process data. Online data is fed to the system leading to continuous training and update of the model. The advantage of this approach is that no manual work is required in the update process. The major disadvantage is that the network may be over-adjusted and as a result, the model will learn the faulty behavior as normal operation and lose fault detection capability. Periodically updated models require a lot of manual work from experienced personnel. This work includes the training procedure, which depending on the software, can be very time-consuming. More importantly, the training period and corresponding training data should be chosen with special care, which requires detailed knowledge of the process and plant events and operation. This in turn requires manual inspection of the data and the resulting personnel cost can be high. In general, the training frequency of periodically updated models and the corresponding personnel costs can have a large proportion of the system upkeep costs.

Component's failure rate usually follows the so-called bathtub curve, see Figure 25, where the breakdown probability of a component is high in the beginning of the plant operation due to design defects and installation errors. The failure rate decreases during the early life period, which is followed by the useful life period when the failure rate remains constant until equipment start to suffer from mechanical stress and wear and the failure rate increases. (Marlow et al. 2017, 716) Often a good starting point to network training is to use data from the early stages of normal operation when process equipment are new and are considered healthy. Another possibility is to update the model after revisions when the process equipment are maintained. It is also clear that after process changes the used model should be retrained. The obvious challenge in these situations is the lack of historical data. Physical process simulators could be used to obtain the required training data but in practice, such excessive and accurate simulators are hardly available. This means that often after process changes the only option is to use the existing ANN model until new training data is available. This needs to be considered when developing a condition monitoring system and for these situations other monitoring methods should be reserved.

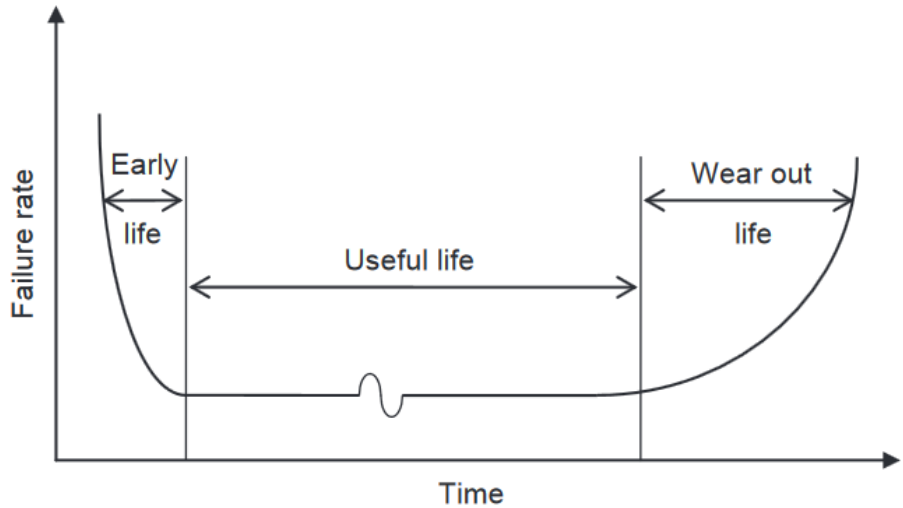


Figure 25. Theoretical bathtub curve (Singh & Adachi, 2013, 107).

The third practical question in model training is the length of the training period and the corresponding size of the training data set. The availability of historical data can set practical limitations to the training period but in general, as illustrated in the case study in Section 6, longer training periods result in better models. Short training periods have the fundamental

issue that they rarely represent all the relevant operating conditions and process behaviors that should be monitored in a complex power plant environment. Short training periods can also lack the required number of samples required for the model to learn the system behavior. Combination of a short training period and a complex model structure is also prone to overfitting. Increasing the training period too much has the drawback that the training may not converge optimally. Increasing the training period also increases the required computational effort, which however, rarely is a limiting factor in condition monitoring applications.

As with other ANN training parameters, the optimal training period length always depends on the monitored system. Systems with simpler dynamics and fewer changing conditions can be successfully modeled with shorter training periods than systems with more complex behavior. The optimal training period also depends on the overall complexity of the network. Models with high degree of freedom resulting from a large independent variable set or large number of hidden neurons/layers should be trained with larger training data sets to avoid overfitting. It is also worthwhile to note that the training period does not need to be continuous. Instead, it is often advisable to select multiple training data sets corresponding to different conditions, e.g. seasons and loads.

The importance of adequate data preprocessing should not be underestimated. Usually the preprocessing is integrated into commercial software packages or at least, the software developer provides instructions and requirements for the preprocessing.

7.1.2 Input variable selection

The second fundamental consideration in the ANN training is the selection of input variables, which has a major impact on the network's performance and accuracy. Three aspects make the input variable selection (IVS) challenging: The number of available variables is usually very large in power processes; many variables have strong correlations with each other; and many variables have weak predictive power. Ideally, the independent variables are selected in a way that the set of variables has high predictive power, the variables are dissimilar to each other with minimal redundancy and no irrelevant variables are present increasing noise in the training. (May et al. 2011, 19)

IVS algorithms in the literature can be categorized in three main classes: wrapper, embedded or filter algorithms (May et al. 2011, 23). Automatized IVS algorithms can be useful in some instances and some algorithms are often included in commercial software packages. In condition monitoring applications, the software user should have major a priori knowledge of the monitored system and some estimation of suitable candidate input variables. This knowledge should be the basis of variable selection, which can be augmented with IVS algorithms. Relying purely on automatized algorithms is a tempting option, especially when a large number of models are used. This can be successful, as in the case study in Section 6, but in general, such algorithms should be used with caution. There is no guarantee of any physical relation between automatically selected input variables and the predicted output variable. A common problem with time series is that an input variable x_1 correlating to the output variable y actually have a mutual dependency on a variable x_2 (Fawcett, 2015), such as time or power output. In such cases, variable x_1 correlates with the output variable but has little generalized predictive power and is usually weak to detect faulty situations.

The selection of independent variables is a challenging task even with sufficient knowledge of the studied system. The variables may be selected in multiple ways depending on the monitored system. One way to choose the independent variables is to introduce a distance measure to each monitored parameter defining how distant measurements upstream and downstream relative to the monitored parameter are included. Nearby measurements should correlate well to the output parameter and reduce noise. The risk of using only measurements very close to the output variable is that the model may not learn all the relevant process dynamics. The choice of how distant measurements are included also depends on the monitoring strategy. If only a small number of models are used, more distant measurements should be included in the models to detect faulty behavior. This is because nearby measurements result in very specific models that are capable of detecting anomalies at a specific location. If a fault has occurred further away from the monitored component the fault still most likely remains unnoticed and another model at that specific location is required.

The number of independent variables has a significant impact on the network's performance. Ideally the independent variable set is kept as small as possible while still preserving predictive power. This way the network's complexity is not unnecessarily increased, and chance

of overfitting is decreased. Also, the iteration of the network's weight coefficients is more likely to converge, and computational effort is decreased. However, larger set of independent variables usually provide more flexibility and possess more predictive power as more features are included in the model, which is often useful for more complex systems. This is visible in the case study results, as models with 30 independent variables clearly outperform models with 10 independent variables.

In order to keep the number of independent variables as small as possible and to decrease the chance of overfitting, redundant variables are to be avoided. Power plants often have several parallel measurements of the same quantity, e.g. temperature or pressure measurements at the same location. These parallel measurements represent the same feature and including them all in the model only increases the model complexity while bringing little or no additional improvement. It is therefore advisable to select only one of the parallel measurements or use their average. Similarly, the set of independent variables should, as the name states, be independent. If the input variables are linearly dependent, the same variable is effectively introduced as multiple inputs. Commercial software often apply a sensitivity analysis to the set of independent variables showing how significant impact each input variable has on to the output variable. This analysis can be used to select the most appropriate parallel measurement and to help choosing the candidate variables with highest predictive power. However, the relation between the input and output variables is very complex to measure and therefore, such algorithms should be treated with care.

7.1.3 Network architecture

The network architecture is defined by the number of hidden layers, hidden neurons in these layers and the connections between the neurons. While it is shown that a multilayer feedforward network with only one hidden layer is capable of approximating any nonlinear function (Hornik 1990), additional hidden layers may be helpful in the model training with complex data sets, such as time-series (Heaton 2017).

Again, the optimal network architecture is problem specific and depends in a complex way on the network type, the complexity of the learning task, neuron activation functions, training algorithm and other training parameters. Some rules-of-thumb for neuron selection have been presented, such as the number of hidden neurons should be less than twice the size of

the input layer (Heaton 2017). Several statistic methods have also been created, some of which are presented in Table 9. Even though such methods have been proposed, they are difficult to implement when using commercial software packages. These methods are specific for certain criteria, require strict a priori assumptions and are still crude approximates at best. Ultimately, finding a working network architecture comes down to trial and error.

Table 9. Various approaches to fix the number of hidden neurons. N_h denotes the number of hidden neurons, n_i the number of inputs, n_o the number of outputs, N the input-target relation, n_p the number of input samples and L the number of hidden layer. Adopted from Sheela & Deepa (2013, 9).

Method	Year	Number of hidden neurons
Li et al. method	1995	$N_h = (\sqrt{1+8n_i} - 1)/2$
Tamura and Tateishi method	1997	$N_h = N - 1$
Zhang et al. method	2003	$N_h = 2^{n_i}/n_i + 1$
Jinchuan and Xinzhe method	2008	$N_h = (n_i + \sqrt{n_p})/L$
Shibata and Ikeda method	2009	$N_h = \sqrt{n_i n_o}$
Sheela and Deepa method	2013	$N_h = (4n_i^2 + 3)/(n_i^2 - 8)$

Often the best network architecture is the simplest one. It is therefore advisable to start model training with a simple architecture. Often a practical starting point is provided by the software's default settings or recommended architecture. If the created model is not satisfactory, the architecture complexity can be progressively increased by adding hidden neurons and/or layers.

7.2 Model verification

Validation and verification of used ANN models is important for a reliable CMS. Validation refers to checking the model performance during training. Validation is dependent on the used training algorithm and is usually automated in commercial software. Verification refers to ensuring the model performance after training and is discussed in this section.

Ideally, every used model should be verified with two additional data sets: healthy state data and faulty state data. Statistical measures can be used to evaluate the model's prediction in comparison to the healthy state data. Some of the most used measures include mean absolute error (MAE), mean squared error (MSE) and root mean squared error (RMSE), whose results are summed over the whole data set and normalized by the size of the data set. The metrics are defined by the following equations:

$$RMSE = \sqrt{\sum_{1=1}^{n}\sum_{j=1}^{m}\left(o_{ij} - z_{ij}\right)^2} \tag{11}$$

$$MSE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(o_{ij} - z_{ij}\right)^2}{n_p} \tag{12}$$

$$MAE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left|o_{ij} - z_{ij}\right|}{n_p} \tag{13}$$

where n_p is the number of patterns in the data set, m is the number of outputs, o is the output of a single neuron j, z is the target for the single neuron j, and i denotes input patterns (Twomey & Smith, 1996). Another useful way to evaluate the model performance is to use scatter plots and calculate the coefficient of determination R^2 as

$$SS_{\text{res}} = \sum_{i}(y_i - f_i)^2 \tag{14}$$

$$SS_{\text{tot}} = \sum_{i}(y_i - \bar{y})^2 \tag{15}$$

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}} \tag{16}$$

where y denotes predicted values and f measured values (Rasila 2007).

The R^2 together with scatter plots provide a simple tool to evaluate the model prediction. However, it should be highlighted, that R^2 is a relative measure for the goodness of fit and should not be used as only performance metric (Reisinger 1997, 1). Figure 26 shows the reference model 2CH prediction vs. measured values during each phase. The high R^2 values indicate very high prediction accuracy. The plots are also useful in determining weak areas

in the model performance. The highest deviation between measured and predicted values occur with low flue gas temperatures. In this case, these values are not interesting, since they correspond to plant downtime, but similar situations may be interesting in other applications. The effect of plant downtime and other similar situations that are not relevant from the perspective of condition monitoring should be taken into account when evaluating the model accuracy. These situations are often not included in the model training. If data periods corresponding to these situations are present in statistical model evaluation, they can skew the performance metrics.

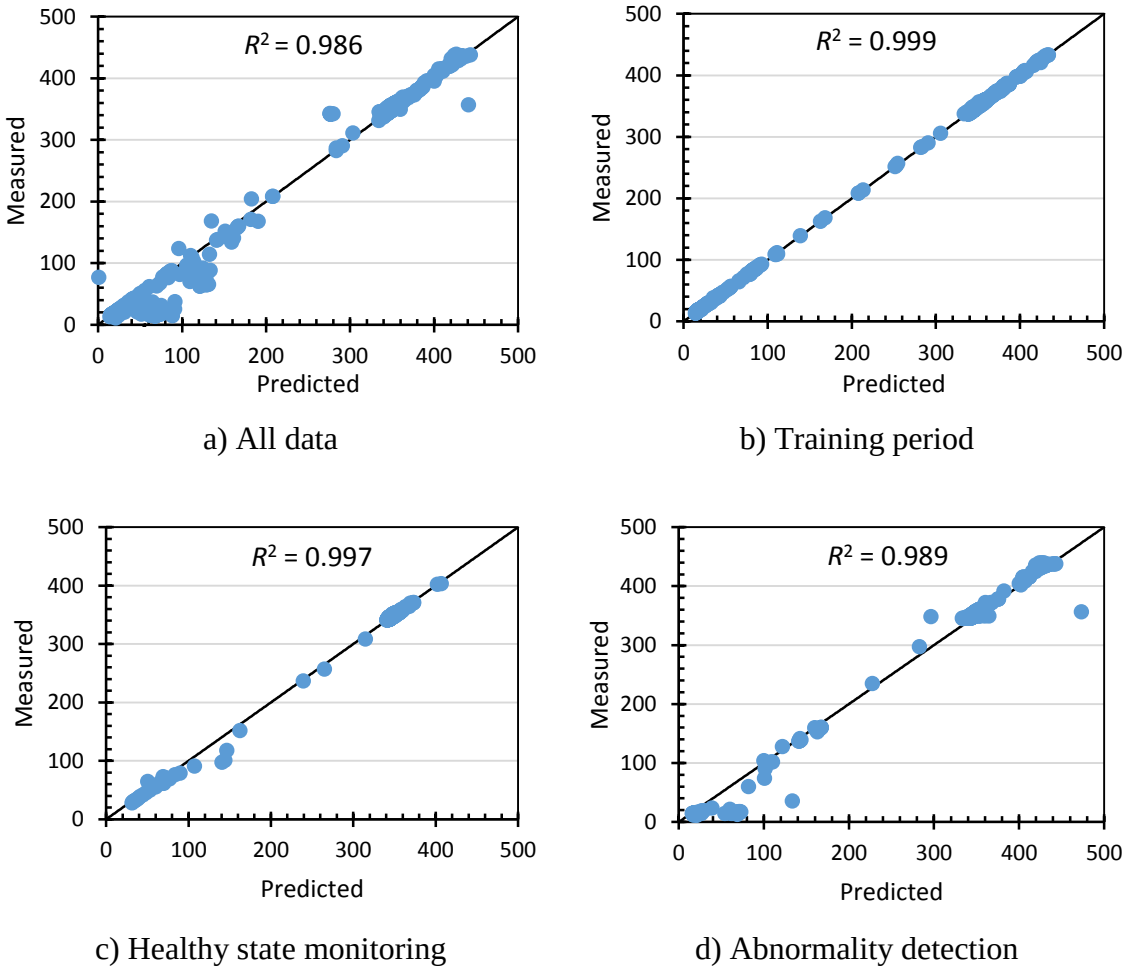


Figure 26. Measured vs. predicted values of the reference model 2CH with very good performance.

Verification of healthy state prediction is relatively simple if only sufficient amount of data is available. Commercial software packages usually include some statistical measures to

evaluate the model performance and the above described performance metrics are also simple to implement using standard tools, such as Microsoft Excel. Another way is to use graphical inspection and linguistic statements, as in the case study in Section 6. This approach is not as objective or precise as statistical measures but is still sufficient enough in many cases. Linguistic statements based on graphical inspection provide often the simplest way to model evaluation, which is desirable when creating a large number of models. The model training typically requires creating many models with different parameters and many personnel may be involved in this process. Linguistic statements also provide the easiest way to share information between involved personnel without the need to interpret any statistical metrics.

The main challenges in the model verification arise from defining the required model performance level and from the lack of faulty state data. Regardless of the performance metrics used, the final decision of defining which level of model accuracy is required, is determined by the software user. The goal of using ANN models in plant-wide condition monitoring is to detect abnormalities during operation and also this capability should be verified. The case study results underline this requirement, as several models had accurate prediction during the training and healthy state operation but were unable to detect the increased flue gas temperature. Verifying this capability can be challenging, since the available data usually lacks clear failures. If a large number of components is monitored, it is practically impossible to verify the fault detection capability of each created model. However, it is highly beneficial to verify this capability with as many components as possible to establish trust in the monitoring approach.

7.3 Model training and verification summary

Finding optimal network structure, independent variables and training parameters is a difficult task that depends on many factors in a complex way. No universal solution exists and the procedure involves many trials and errors. In plant-wide condition monitoring, the actual task is often not finding an optimal ANN model. In most cases, it is enough to find a model accurate enough to detect abnormalities without excessive false alarms during healthy state operation for the operators to find the model reliable. To establish this trust, each used model should be verified with healthy state data and at least one model with faulty state data.

It is vital to understand that the training parameters depend on each other, e.g. using a large number of hidden neurons typically requires large training data set. It is also worth to notice that a reliable model can be trained with several different parameter combinations, but on the other hand, there is no guarantee the same parameters result in an accurate model on some other component. In addition, information flow of detailed plant events should be provided to the software user to assist in the training period selection to avoid introducing faulty state operation in the training phase.

As the training process is iterative by nature and the aim is to monitor a wide range of equipment, it is important to create general guidelines for the model training with a specific software. Initial systematic in-house testing with different network structures and training parameters should be carried out to determine a practical starting point defining initial training parameters for the model training. These initial training parameters can be used in the proposed general flowchart presented in Figure 27. Documented testing and acquired experience will significantly reduce the workload when monitoring is extended to additional plants.

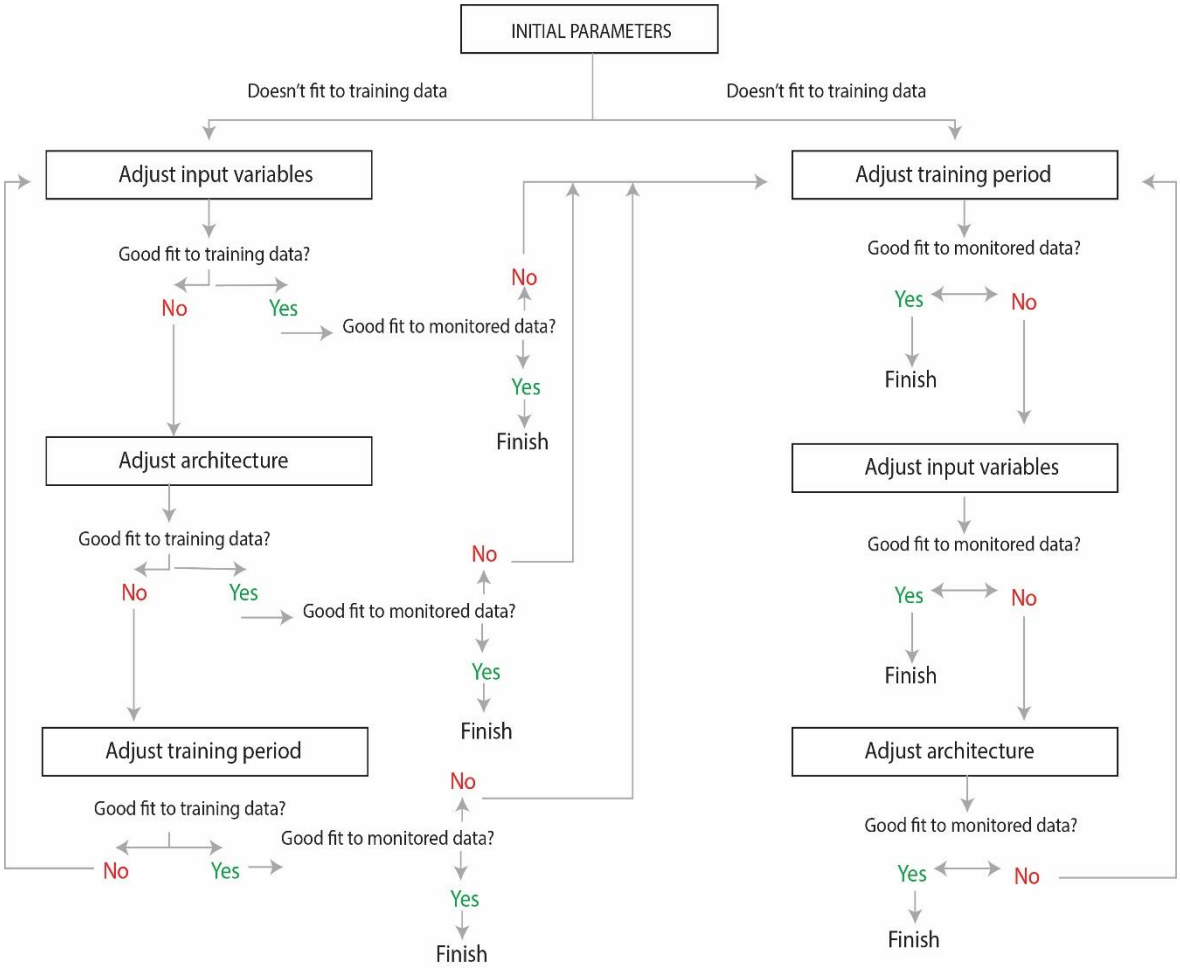


Figure 27. General flowchart for ANN model training.

8 CONCLUSIONS

Power plant maintenance has been traditionally organized according to corrective and preventive maintenance strategies, which both have their limitations that can lead to high costs due to low availability and inefficiently timed maintenance actions. CBM is considered an advanced maintenance strategy and is already applied in many critical components in modern power plants. Extending condition monitoring and more efficient maintenance practices to a wider range of equipment is desirable but requires advanced condition monitoring methods that can detect changes in process parameters indicating faults and malfunctions. These methods include different mathematical models that can be classified as physics-based, data-driven and hybrid models.

Artificial neural networks are a popular data-driven approach due to their self-learning capability and their ability to approximate nonlinear functions. ANN's can be used in many ways in condition monitoring applications including pattern recognition, classification and time series prediction. Many commercial condition monitoring software based on ANN's are available for asset owners and O&M service providers. In this thesis, a commercial software Intelligent Health Monitor (algorithmica technologies GmbH) was evaluated in a case study to monitor a boiler's increased flue gas temperature, which indicates decreased performance of heat surfaces prior to economizer. The software is based on recursive neural networks that are trained with historical data from healthy state operation.

A total of 96 models were created to monitor historical data of the flue gas temperature. Each model was trained using different parameters and evaluated based on graphical inspection of the model trend. The varied parameters were the length of the training data period, the network architecture including the number of hidden neurons and layers, and the independent variables used to predict the output variable. The historical data of three years was divided into three periods: training period, healthy state operation and faulty state operation when the flue gas temperature had gradually increased. 13 of the 96 models were considered reliable, as they were able to accurately predict the healthy state operation during training and monitoring phases and also detect the increased flue gas temperature. In general, the results of the case study significantly varied but two conclusions based on the results can be made: training period of one month is too short for a reliable model and 30 automatically chosen independent variables based on PCA and correlation analysis led to best performing models.

The case study showed that ANN based software can be successfully used to detect abnormalities in power plant environment. However, the case study only includes the monitoring of a single parameter and the applicability to other monitored parameters should be studied further. The case study also showed that the network training and verification are case-specific and can be very challenging. The following aspects should be considered in the model training and implementation to a condition monitoring system.

1. No universal solution for optimal network structure or parameter selection exists. The training parameters depend on many factors in a complex way, e.g. system complexity, network complexity, available training samples, and used training algorithm and the training parameters also affect each other. In general, lots of trials and errors are to be expected in the model training.

2. Systematic in-house testing is advisable to determine practical initial training parameters for a specific software. Power plants have a range of similar components and often asset owners and O&M service providers monitor several plants. Well-documented testing can provide a practical starting point for model training, which can significantly reduce the work load when extending condition monitoring to additional plants.

3. Each used model should be verified with additional data set corresponding to healthy state operation. In addition, the fault detection capability should be verified, since accurate healthy state prediction does not guarantee that the model can recognize abnormal situations. This verification can be challenging, since such data sets are often difficult to obtain.

The research objective of this thesis was met. The theoretical part gives a broad overview of power plant maintenance and related modelling methods providing a fundamental introduction to the topic. The case study results and presented conclusions give valuable information for O&M service providers and asset owners seeking new approaches for more advanced condition monitoring. While ANN based condition monitoring is an attractive option for plant-wide monitoring, many stakeholders hesitate to implement ANN based condition monitoring into their condition monitoring systems due to the black-box nature of ANNs and due to the fact that hardly any objective evaluation of commercial software are available. This work bridges the gap between research and practice by offering practical considerations to the implementation of ANN based condition monitoring and by providing documented software evaluation.

8.1 Future work

Similar testing as in the case study should be carried out with other monitored parameters. The case study showed that 30 automatically chosen independent variables, training period of four months and hidden neurons arranged as 45-22-11-5 in four hidden layers resulted in an accurate model. These same initial training parameters should be tested with other monitored parameters to see if they can be used as a practical starting point in the model creation.

Adequate training and verification data are often difficult to obtain. One way to obtain such data sets is to use physics-based process simulators and this option should be studied. Another practical research topic is the utilization of different modelling methods in condition monitoring systems. O&M service providers require tools for different levels of condition monitoring and the combined use of different software could provide flexible practical solutions for condition monitoring in power plants.

REFERENCES

Akula Aparna, Goel Sudhanshu, Ghosh Ripul. 2017. Condition Monitoring Saves Money and Prevents Failures. Chemical Engineering Progress. Vol. 113, Iss. 8, pp. 58–62.

Al-Turki Umar, Ayar Tahir, Yilbas Bekir Sami, Sahin Ahmet Ziyaettin. 2017. Integrated Maintenance Planning in Manufacturing Systems. Cham: Springer. ISBN 978-3-319-06289-1.

Algorithmica technologies GmbH. Intelligent Health Monitor. 2018. [Online]. [Accessed on 26.3.2018]. Available: http://www.algorithmica-technologies.com/en/solutions/intelligent-health-monitor

Alman David H. & Ningfang Liao. 2001. Overtraining in Back-Propagation Neural Networks: A CRT Color Calibration Example. Color Research and Application, Vol. 27, No. 2, pp. 122–125.

Bach Francis R. 2014. Breaking the Curse of Dimensionality with Convex Neural Networks. eprint arXiv:1412.8690

Bangert Patrick. 2012. Optimization for Industrial Problems. Heidelberg: Springer. ISBN 978-3-642-24973-0.

Bangert Patrick. 2018. Smart Condition Monitoring Using Machine Learning. algorithmica technologies GmbH. [Online]. [Accessed on 23.11.2017]. Available: http://www.algorithmica-technologies.com/system/case_studies/pdf_ens/000/000/022/original/16_Smart_Condition_Monitoring.pdf?1498681861

Bania R.K. 2014. Survey on Feature Selection for Data Reduction. International Journal of Computer Applications. Vol. 94, No. 18.

Bengtsson Markus, Olsson Erik, Funk Peter & Jackson Mats. 2004. Technical Design of Condition Based Maintenance System – A Case Study using Sound Analysis and Case-Based Reasoning. Proceedings of the 8th Conference of Maintenance and Reliability.

Bousdekis Alexandros, Magoutas Babis, Apostolou Dimitris & Mentzas Gregoris. A proactive decision making framework for condition-based maintenance. Industrial Management & Data Systems, Vol. 115 Iss. 7, pp. 1225–1250.

Chanda Pradip & Mukhopaddhyay Suparna. 2016. Operation and Maintenance of Thermal Power Stations. India: Springer. ISBN 978-81-322-2720-5.

Chen Jianhong, Li Honkung, Sheng Deren & Li Wei. 2015. A hybrid data-driven modeling method on sensor condition monitoring and fault diagnosis for power plants. International Journal of Electrical Power & Energy Systems. Vol. 71, pp. 274–284.

Chookah M., Nuhi M. & Modarres M. A probabilistic physics-of-failure model for prognostic health management of structures subject to pitting and corrosion-fatigue. Reliability Engineering & System Safety. Vol. 96, Iss. 12, pp. 1601–1610.

Crespo Adolfo Márquez. 2007. The Maintenance Management Framework. London: Springer. ISBN 978-1-84628-820-3.

Das Anupam, Maiti J. & Banerjee R.N. 2012. Process monitoring and fault detection strategies: a review. International Journal of Quality & Reliability Management, Vol. 29, Issue 7, pp. 720–752.

Di Maio Francesco, Colli Davide, Zio Enrico, Tao Liu & Tong Jiejuan. 2015. A Multi-State Physics Modeling approach for the reliability assessment of Nuclear Power Plants piping systems. Annals of Nuclear Energy. Vol 80, pp. 151–165.

El-Thalji Idriss & Jantunen Erkki. 2015. A summary of fault modelling and predictive health monitoring of rolling element bearings. Mechanical Systems and Signal Processing, Vol. 60–61, pp. 252–272.

EN 13306:2001. 2001. Maintenance terminology. European Standard. CEN (European Committee for Standardization), Brussels.

Fawcett Tom. 2015. Avoiding Common Mistakes with Time Series. Silicon Valley Data Science. [Online]. [Accessed on 18.3.2018]. Available: https://svds.com/avoiding-common-mistakes-with-time-series/

Fedele Lorenzo. 2011. Methodologies and Techniques for Advanced Maintenance. London: Springer. ISBN 978-0-85729-102-8.

Fortum. 2017a. Annual Review 2016. [Online]. [Accessed on 20.1.2018]. Available: http://annualreport2016.fortum.com/en/

Fortum 2017b. Performance, OM and energy efficiency services for thermal power plants. [Online]. [Accessed on 20.1.2018]. Available: https://www3.fortum.com/products-and-services/power-plant-services/services-thermal-power-plants/performance-om-and-energy

Frisk Erik. 2001. Residual Generation for Fault Diagnosis. Doctoral thesis. Linköping University. Department of Electrical Engineering. ISBN 91-7373-115-3.

Fuentes-Garcia Marta, Macia-Fernandez Gabriel & Camacho Jose. 2018. Evaluation of diagnosis methods in PCA-based Multivariate Statistical Process Control. Chemometrics and Intelligent Laboratory Systems. Vol. 172, pp. 194–210.

Garcia Salvador, Luengo Julian & Herrera Francisco. 2015. Data Preprocessing in Data Mining. Cham: Springer. ISBN 978-3-319-10246-7.

Garcia-Matos Jesus, Sanz-Bobi Miguel, Munoz Antonio, Sola Antonio. 2013. Hybrid Model-Based Fault Detection and Diagnosis for the Axial Flow Compressor of a Combined-Cycle Power Plant. Journal of Engineering for Gas Turbines and Power. Transactions of the ASME. Vol. 135, Iss. 5, pp. 1–5.

Ge Zhiqiang. 2017. Review on data-driven modeling and monitoring for plant-wide industrial processes. Chemometrics and Intelligent Laboratory Systems 171, pp. 16–25.

Goofellow Ian, Bengio Yoshuaa and Courville Aaron. 2016. Deep Learning. MIT Press.

Gräber Ulrich. 2004. Advanced Maintenance Strategies for Power Plant Operators – Introducing Inter-plant Life Cycle Management. International Journal of Pressure Vessels and Piping 81, pp. 861–8625

Guillén A.J. Cresp A. Macchi M. & Gómez J. 2016. On the role of Prognostics and Health Management in Advanced Maintenance Systems. Production Planning & Control Vol. 27, No. 12, pp. 991–1004.

Haddad Gilbert, Sandborn Peter & Pecht Michael. 2012. An Options Approach for Decision Support of Systems With Prognostics Capabilities. IEEE Transactions on Reliability Vol. 61, No. 4, pp. 872–883.

Hashemian H.M. & Bean Wendell C. State-of-the-Art Predictive Maintenance Techniques, in IEEE Transactions on Instrumentation and Measurement, vol. 60, no. 10, pp. 3480–3492.

Heaton Jeff. 2017. The Number of Hidden Layers. Heaton Research. [Online]. [Accessed on 12.3..2018]. Available at: http://www.heatonresearch.com/2017/06/01/hidden-layers.html

Hira Zena M. & Gillies Duncan F. 2015. A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data. Advances in Bioinformatics, 2015, 198363.

Holmberg Kenneth, Adgar Adam, Arnaiz Aitor, Jantunen Erkki, Mascolo Julien, Mekid Samir. 2010. E-maintenance. London: Springer. ISBN 978-1-84996-205.

Hornik Kurt. 1990. Approximation Capabilities of Multilayer Feedforward Networks. Neural networks, Vol. 4, pp. 251–257.

Huhtinen Markku, Kettunen Arto, Nurminen Pasi & Pakkanen Heikki. 1994. Höyrykattila-tekniikka. Opetushallitus: Helsinki. ISBN 951-37-1327-X.

Hush D.R., Abdallah C.T., Heileman G.L. & Docampo D. 1997. Neural Networks in Fault Detection: A Case Study. Proceedings of the American Control Conference. Vol 2, pp. 918–921.

Isermann Rolf. 2006. Fault-Diagnosis Systems. Springer: Berlin. ISBN 978-3-540-24112-6.

Ismail Rahmat Izaibi B., Alnaimi Firas B. Ismail & AL-Qrimli Haidar F. 2016. Artificial Intelligence Application in Power Generation Industry: Initial considerations. International Conference on Advances in Renewable Energy and Technologies. IOP Conf. Series: Earth and Environmental Science 32 (2016).

Jaako Juha. 1998. Aspects of process modelling. University of Oulu. Control Engineering Laboratory. ISBN 951-42-5035-4.

Jardine Andrew, Lin Daming & Banjevic Dragan. 2006. A Review on Machinery Diagnostics and Prognostics Implementing Condition-Based Maintenance. Mechanical Systems and Signal Processing, Vol. 20, pp. 1483–1510.

Kaija Matti. 2016. Utilization of condition monitoring in power plant operation and maintenance services. Master of Science Thesis. Tampere University of Technology.

Khoukhi Amir & Khalid Mohamed. 2015. Hybrid computing techniques for fault detection and isolation, a review. Computers and Electrical Engineering. Vol 43, pp. 17–32.

Kobbacy Khairy & Murthy Prabhakar. 2008. Complex System Maintenance Handbook. London: Springer. ISBN 978-1-84800-010-0.

Kylliäinen Ville-Veikko, Laaksonen Leila & Viitasaari Osmo. 2011. Käytettävyys, kunto ja elinikä – kunnossapitostrategia tuottavuuden parantamisessa. Promaint. 5/2011, pp. 10–15.

Largueh Samia, Aloui Sinda, Pages Oliver & Chaari Abdessattar. 2015. Fuzzy Sliding Mode Control For Turbocharger Diesel Engine. Journal of Dynamic Systems, Measurement, and Control. 138.

Leite Gustavo de Novaes Pires, Araújo Alex Maurício & Rosas Pedro André Carvalho. 2018. Prognostic techniques applied to maintenance of wind turbines: a concise and specific review. Renewable and Sustainable Energy Reviews, Vol. 81, Part 2, pp. 1917–1925.

Leturiondo Urko. 2016. Hybrid Modelling in Condition Monitoring. Doctoral thesis. Luleå University of Technology. Operation and Maintenance. ISBN 978-91-7583-722-2.

MacGregor John & Cinar Ali. 2012. Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods. Computers and Chemical Engineering. Vol. 47, pp. 111–120.

Manning-Ohren David. 2015. Top ten rules for power plant condition monitoring. Power Engineering International [Online]. [Accessed on 28.9.2017]. Available: http://www.powerengineeringint.com/articles/print/volume-23/issue-5/features/top-ten-rules-for-condition-monitoring.html

Mansouri Majdi, Mohammed Z.S., Baklouti Raoudha, Nounou Mohamed, Nounou Hazem, Hamida Ahmed Ben & Karim Nazmul. 2016. Statistical fault detection of chemical process – comparative studies. Journal of Chemical Engineering & Process Technology, Vol. 7, Iss. 1.

MANTIS. 2015. About MANTIS. [Online]. [Accessed on 26.1.2018]. Available: http://www.mantis-project.eu/about-mantis/.

Matignon Randall. 2005. Neural Network Modeling Using Sas Enterprise Miner. AuthorHouse. ISBN 1-4184-6675-1.

Marlow D.O., Sanchez S.M. & Sanchez P.J. 2017. Assessing the impact of the bathtub curve failure rate on fleet performance using designed simulation experiments. 22nd International Congress on Modelling and Simulation, Hobart, Tasmania, Australia, 3 to 8 December 2017.

Martha de Souza Gilberto Francisco. 2012. Thermal Power Plant Performance Analysis. Dordrecht: Springer. ISBN 978-1-4471-2309-5.

Marwala Tshilidzi. 2012. Condition Monitoring Using Computational Intelligence Methods. London: Springer. ISBN 978-1-4471-2379-8. J. Chem. Eng. Process Technol. Vol. 7, no 1, pp. 1–10.

May Robert, Dandy Graeme, Maier Holger & Suzuki Kenji (Ed.). 2011. Review of Input Variable Selection Methods for Artificial Neural Networks – Methodological Advances and Biomedical Applications. InTech. ISBN: 978-953-307-243-2.

Mendel Jerry M. 1995. Fuzzy Logic Systems for Engineering: A Tutorial. Proceedings of the IEEE. Vol. 83(3), pp. 345–377.

Mobley Keith R. 2002. An Introduction to Predictive Maintenance. 2nd Edition. Elsevier Science. ISBN 0-7506-7531-4.

Moubray John. 1997. Reliability-centered Maintenance. 2nd Edition. New York: Industrial Press Inc. ISBN 0-8311-3078-4.

Ndjenja L. & Visser J.K. 2015. Development of a Maintenance Strategy for Power Generation Plants. South African Institute of Electrical Engineers. Vol. 106, pp. 132–140.

Niu Gang. 2017. Data-Driven Technology for Engineering Systems Health Management. Beijing: Springer. ISBN 978-981-10-2031-5.

Niu Gang, Yang Bo-Suk & Pecht Michael. 2010. Development of an optimized condition-based maintenance system by data fusion and reliability-centered maintenance. Reliability Engineering and System Safety. Vol 95, pp. 786–796.

Opetushallitus. Värähtelymittaukset [Online]. [Accessed on 20.9.2017]. Available: http://www03.edu.fi/oppimateriaalit/kunnossapito/mekaniikka_k2_varahtelymit-taukset.html

Patan Krzysztof. 2008. Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes. Verlag: Springer. ISBN 978-3-540-79871-2.

Pecht Michael & Gu Jie. 2009. Physics-of-failure-based prognostics for electronic products. Transactions of the Institute of Measurement and Control, Vol. 31, Iss. 3–4, pp. 309–322.

Pecht Michael & Jaai Rubyca. 2010. A prognostics and health management roadmap for information and electronics-rich systems. Microelectronics Reliability 50, pp. 317–323.

Pusey Henry C. 2007. Turbomachinery Condition Monitoring and Failure Prognosis. Sound And Vibration March 2007, pp. 8–12.

Qin Joe S. 2012. Survey on data-driven industrial process monitoring and diagnosis. Annual Reviews in Control. Vol. 36, pp. 220–234.

Qiu Tian & Bai Xiaojing. 2012. Comparison of Different Fault Detection Statistics Detectability in PCA. The 2nd International Conference on Computer Application and System Modeling.

Radhakrishnan Pranoy. 2017. Introduction to Recurrent Neural Network. Towards Data Science. [Online]. [Accessed on 14.3.2018]. Available: https://towardsdatascience.com/introduction-to-recurrent-neural-network-27202c3945f3

Rasila Antti. 2007. Sovellettu todennäköisyyslaskenta B. Aalto University. [Online]. [Accessed on 18.3.2018]. Available: http://www.mv.helsinki.fi/home/mmattila/kvanti/luento8.pdf

Reisinger Heribert. 1997. The impact of research designs on R^2 in linear regression models: an exploratory meta-analysis. Journal of Empirical Generalisations in Marketing Science, Vol. 2.

Roulias Dimitrios, Louats Theodoros & Kostopoulos Vassilios. 2013. A Statistical Feature utilising Wavelet Denoising and Neighblock Method for Improved Condition Monitoring of Rolling Bearings. Chemical Engineering Transactions. Vol. 3, pp. 1045–1050.

Severson Kristen, Chaiwaranodom Paphonwit & Braatz Richard D. 2016. Perspectives on process monitoring of industrial systems. Annual Reviews in Control. Vol. 42, pp. 190–200.

Shanmuganathan Subana. 2016. Artificial Neural Network Modelling: An Introduction. Book Section in: Shanmuganathan Subana (edit) & Sandhya Samarasinghe (edit). Artificial Neural Network Modelling, pp 1–14. Switzerland: Springer. ISBN 978-3-319-28493-4.

Sheriff Mohammed, Botre Chiranjivi, Mansouri Majdi, Nounou Hazem, Nounou Mohamed & Karim Mohammad. 2017. Process Monitoring Using Data-Based Fault Detection Techniques: Comparative Studies. Book Section in: Demetgul Mustafa (edit.) & Ünal Muhammet (edit.). Fault Diagnosis and Detection, pp. 238-261. InTech. ISBN 978-953-51-3204-2.

Shin Jong-Ho & Jun Hong-Bae. 2015. On condition based maintenance policy. Journal of Computational Design and Engineering, Vol. 2, pp. 119–127.

Simani S. & Fantucci C. 2000. Fault diagnosis in power plant using neural networks. Information Sciences. Vol. 127, Iss. 3–4, pp. 125–136.

Singh Amarjit & Adachi Stacy. 2013. Bathtub curves and pipe prioritization based on failure rate. Built Environment Project and Asset Management, Vol. 3 Iss. 1, pp.105-122.

Sobhani-Tehrani Ehsan & Khorasani Khashayar. 2009. Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach. Springer: Dordrecht. ISBN 978-0-387-92906-4.

Song Yong-Hua & Johns Allan T. 1997. Applications of fuzzy logic in power systems. I. General introduction to fuzzy logic. Power Engineering Journal. Vol. 11, Iss. 5, pp. 219–222.

Stanley Greg. 2010. Model Based Reasoning for Fault Detection and Fault Diagnosis. [Online]. [Accessed on 30.10.2017], Available: https://gregstanleyandassociates.com/whitepapers/FaultDiagnosis/Model-Based-Reasoning/model-based-reasoning.htm

Stefaniak Pawel, Zimrow Radoslaw, Sliwinski Pawel, Andrzejewski Marek & Wylomanska Agnieszka. 2014. Multidimensional Signal Analysis for Technical Condition, Operation and Performance Understanding of Heavy Duty Mining Machines. In: Chaari Fakher, Zimroz Radoslaw, Bartelmus Walter & Haddar Mohammed (editors). Advances in Condition Monitoring Machinery in Non-Stationary Operations. Cham: Springer. pp. 197–210. ISBN 978-3-319-20462-8.

Stringer David, Sheth Pradip & Allaire Paul. 2012. Physics-based modeling strategies for diagnostic and prognostic application in aerospace systems. Journal of Intelligent Manufacturing, Vol. 23, Issue 2, pp. 155–162.

Suganthi L., Inyan S. & Samuel Anand A. 2015. Applications of fuzzy logic in renewable energy systems – A review. Renewable and Sustainable Energy Reviews. Vol. 46, pp. 585–607.

Sullivan G.P., Pugh R., Melendez A.P. & Hunt W.D. 2010. Operations and Maintenance Best Practices Guide, Release 3.0. U.S. Department of Energy. Federal Energy Management Program.

Sutharssan Thamo, Stoyanov Stoyan, Bailey Chirs & Yin Chunyan. 2015. Prognostic and health management for engineering systems: a review of the data-driven approach and algorithms. The Journal of Engineering. July 2015.

Tahan Mohammadreza, Tsoutsanis Elias, Muhammad Masdi & Karim Z.A. Abdul. 2017. Performance-based health monitoring, diagnostics and prognostics for condition-based maintenance of gas turbines: A review. Applied Energy, Vol. 198, pp. 122–144.

Talebi Heidar A., Abdollahi Farzaneh, Patel Rajni V. & Khorasani Khashayar. 2010. Neural Network-Based State Estimation of Nonlinear Systems – Application to Fault Detection and Isolation. New York: Springer. ISBN 978-1-4419-1437-8.

Tautz-Weinert J. & Watson S.J. 2017. Combining Model-based Monitoring and a Physics of Failure Approach for Wind Turbine Failure Detection. Presented at the 30th Conference on Condition Monitoring and Diagnostic Engineering Management (COMADEM 2017), University of Central Lancashire, UK, 10-13th July.

Temsamani A.B., Kauffmann S., Descas Y., Vandevelde B., Zanon F. & Willems G. 2017. Improved and accurate physics-of-failure (PoF) methodology for qualification and lifetime assessment of electronic systems. Microelectronics Reliability. Vol. 76–77, pp. 42–46.

Tetko Igor V., Livingstone David J. & Luik Alexander I. Neural Network Studies. 1. Comparison of Overfitting and Overtraining. J: Chem. Inf. Comput. Sci., Vol. 35, No 5, pp. 826–833.

The World Energy Council, 2016. World Energy Recourses 2016. ISBN 978 0 946121 58 8.

Thurston Michael & Lebold M. 2001. Open Standards for Condition-Based Maintenance and Prognostic Systems. 5th Annual Maintenance and Reliability Conference (MARCON 2001), Galtinburg, TN, May 25–27.

Tidriri Khaoula, Chatti Nizar, Verron Sylvain & Tiplica Teodor. 2016. Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges. Annual Reviews in Control. Vol. 42, pp. 63–81.

Twomey Janet M. & Smith Alice E. Validation and Verification. Chapter in: Kartam Nabil, Flood Ian & Garrer James H. 1997. Arftificial Neural Networks for Civil Engineers: Fundamentals and Applications. New York: ASCE Press. ISBN 0784402256.

Tzafestas S.G., Dalianis P.J. % Anthopoulos G. 1996. On the overtraining phenomenon of backpropagation neural networks. Mathematics and Computers in Simulation. Vol 40, pp. 507–521.

Urata Mikiyasu, Aoki Kiyotaka, Urata Naoya, Tagami Koichi, Kondo Manabu & Matsumoto Shintaro. 2015. Non Destructive Inspection Technology for Thermal Power Plants. Mitsubishi Heavy Industries Technical Review. Vol. 52 No. 4, pp. 37–42.

Van Tung Tran & Yang Bo-Suk. 2009. Machine Fault Diagnosis and Prognosis: The State of The Art. International Journal of Fluid Machinery and Systems. Vol. 2, pp. 61–71.

Venkatasubramanian Venkat, Rengaswamy Raghunathah, Yin Kevin & Kavuri Surya. 2003a. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. Computers and Chemical Engineering. Vol 27, Iss. 3, pp. 293–311.

Venkatasubramanian Venkat, Rengaswamy Raghunathah, Yin Kevin & Kavuri Surya. 2003b. A review of process fault detection and diagnosis: Part III: Process history based methods. Computers and Chemical Engineering. Vol 27, Iss. 3, pp. 327–346.

Villegas Thamara, Fuente Maria Jesus & Rodriguez Miguel. Principal Component Analysis for Fault Detection and Diagnosis. Experience with a pilot plant. International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics - Proceedings 2010. pp. 147–152.

Westerhuis Johan, Gurden Stephen & Smilde Age. 2000. Generalized contribution plots in multivariate statistical process monitoring. Chemometrics and Intelligent Laboratory Systems. Vol. 51, Iss. 1, pp. 95–114.

Wireman Terry. 2004. Total Productive Maintenance. 2nd Edition. New York: Industrial Press Inc. ISBN 0-8311-3172-1.

Yan Jihong. 2014. Machinery Prognostics and Prognosis Oriented Maintenance Management. Singapore: John Wiley & Sons. ISBN 111863876X.

Yang Qingsong. 2004. Model-Based and Data Driven Fault Diagnosis Methods with Applications to Process Monitoring. Doctoral thesis. Case Western Reserve University. Electrical Engineering and Computer Sciences.

Zhang Q.J. & Gupta K.C. 2000. Neural Networks for RF and Microwave Design. Artech House, Inc. Norwood, MA, USA. ISBN 1580531008.

Zhang Wan, Jina Min-Ping, Zhu Lin & Yan Xiao-An. 2017. Comprehensive Overview on Computational Intelligence Techniques for Machinery Condition Monitoring and Fault Diagnosis. Chinese Journal of Mechanical Engineering. Vol. 30 (4), pp. 782–795.

Zhang Zhenyou, Wang Yi & Wang Kesheng. 2013. Intelligent fault diagnosis and prognosis approach for rotating machinery integrating wavelet transform, principal component analysis, and artificial neural networks. The International Journal of Advanced Manufacturing Technology. Vol. 68, Iss. 1–4, pp. 762–773.

Zhou Shaoyuan, Zhang Jianming & Wang Shuquing. 2004. Fault Diagnosis in Industrial Processes Using Principal Component Analysis and Hidden Markov Model. Proceedings of the 2004 American Control Conference. Vol. 6.

Zhu Shun-Peng, Huang Hong-Zhong, Peng Weiwen, Wang Hai-Kun & Mahadevan Sankaran. 2016. Probabilistic Physics of Failure-based framework for fatigue life prediction of aircraft gas turbine discs under uncertainty. Reliability Engineering & System Safety. Vol. 146, pp. 1–12.