Artificial Neural Networks for Gas Turbine Monitoring

Magnus Fast

December 2010
Doctoral Thesis
Division of Thermal Power Engineering
Department of Energy Sciences
Faculty of Engineering
Lund University
Sweden
Abstract

Due to the deregulation of the electricity market the power producers are forced to continuously investigate various means of maintaining/increasing their profits. Improving the electrical efficiency through hardware upgrades is probably the most commonly employed measure, although the interest for enhancements with regard to plant operation is on the rise. Plant operation improvement is often measured in RAM (reliability, availability and maintainability) which acts as an indication of how well a plant can be utilized.

The availability can be increased by employing various monitoring tools allowing the maintenance to be based on the condition rather than equivalent operating hours, thereby extending the periods between overhauls. The reliability can also be increased by employing a combination of monitoring tools alerting the plant operators before faults are fully developed.

Modern power plants are equipped with distributed control systems delivering data to the control room through a considerable number of sensors. This data enables the development of data-driven methods for tasks such as condition monitoring, diagnosis and sensor validation. Artificial neural networks have proven suitable for the non-linear modeling of power plants and its components, and represent the data modeling tools used in this research.

Some of the results of the case studies are very accurate ANN models for different types of gas turbines. Furthermore, the integration of these models and the development of user interfaces for online condition monitoring have been demonstrated.
# Table of Contents

1  Introduction.............................................................................................................. 1
   1.1  Background............................................................................................................ 2
   1.2  Objectives ............................................................................................................ 3
   1.3  Limitations............................................................................................................ 3
   1.4  Methods............................................................................................................... 4
   1.5  Outline of the thesis............................................................................................. 5
   1.6  Acknowledgements............................................................................................... 5

2  Artificial intelligence................................................................................................ 7
   2.1  Artificial neural network....................................................................................... 11
       2.1.1  The artificial neuron..................................................................................... 13
       2.1.2  The single-layer feed-forward network...................................................... 15
       2.1.3  The multi-layer feed-forward network....................................................... 18
   2.2  Considerations when working with MLPs........................................................... 20
       2.2.1  Setup and training...................................................................................... 21
       2.2.2  Data management...................................................................................... 25
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Gas turbine monitoring</td>
<td>27</td>
</tr>
<tr>
<td>3.1</td>
<td>Background</td>
<td>27</td>
</tr>
<tr>
<td>3.2</td>
<td>Condition monitoring</td>
<td>29</td>
</tr>
<tr>
<td>3.3</td>
<td>Fault detection and isolation</td>
<td>31</td>
</tr>
<tr>
<td>3.4</td>
<td>Sensor validation</td>
<td>33</td>
</tr>
<tr>
<td>4</td>
<td>Concluding remarks</td>
<td>37</td>
</tr>
<tr>
<td>5</td>
<td>Summary of Papers</td>
<td>39</td>
</tr>
<tr>
<td>5.1</td>
<td>Papers outside the thesis</td>
<td>43</td>
</tr>
<tr>
<td>6</td>
<td>Bibliography</td>
<td>45</td>
</tr>
</tbody>
</table>
Nomenclature

d Desired output or target -
E Error criterion -
e Error -
F Transfer function -
H Number of hidden neurons -
k Arbitrary neuron -
M Number of input nodes -
N Number of output neurons -
n Iteration number -
s Effective input -
w Weight -
w Weight vector -
x Input vector -
y Input or output signal form neuron -

GREEK
\( \delta \) Delta term or local gradient -
\( \eta \) Learning rate factor -
\( \Sigma \) Summation function -

SUPERSCRIPTS
T Transposed (vector)
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AANN</td>
<td>Auto-associative neural network</td>
</tr>
<tr>
<td>ADALINE</td>
<td>Adaptive linear element</td>
</tr>
<tr>
<td>AI</td>
<td>Artificial intelligence</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial neural network</td>
</tr>
<tr>
<td>ASME</td>
<td>American society of mechanical engineers</td>
</tr>
<tr>
<td>CUSUM</td>
<td>Cumulative sum</td>
</tr>
<tr>
<td>ECOS</td>
<td>Efficiency, cost, optimization and simulation</td>
</tr>
<tr>
<td>EOH</td>
<td>Equivalent operating hours</td>
</tr>
<tr>
<td>FOD</td>
<td>Foreign object damage</td>
</tr>
<tr>
<td>GPA</td>
<td>Gas path analysis</td>
</tr>
<tr>
<td>KBS</td>
<td>Knowledge based system</td>
</tr>
<tr>
<td>LMS</td>
<td>Least mean square</td>
</tr>
<tr>
<td>MLP</td>
<td>Multi-layer perceptron</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean squared error</td>
</tr>
<tr>
<td>OEM</td>
<td>Original equipment manufacturer</td>
</tr>
<tr>
<td>RAM</td>
<td>Reliability, availability and maintainability</td>
</tr>
<tr>
<td>SGC</td>
<td>Swedish gas centre</td>
</tr>
<tr>
<td>SGT</td>
<td>Siemens gas turbine</td>
</tr>
<tr>
<td>TIT</td>
<td>Turbine inlet temperature</td>
</tr>
</tbody>
</table>
This thesis sums up a series of smaller research projects, funded by the Swedish Gas Centre (SGC) and the Thermal Engineering Research Institute (Värmeforsk), on the topic of gas turbine-based power plant monitoring by means of artificial neural networks (ANN). The major goal of these research projects has been to study, develop and evaluate various neural network approaches for condition monitoring, diagnosis and sensor validation.

A gas turbine model can for instance be constructed on the basis of physical laws describing the system behavior. Such an approach is known as mechanistic modeling and involves the formulation of the governing equations such as balance equations for mass, energies and impulses. These models are complex since they include differential equations of the studied system and require much effort in order to become accurate. Especially gas turbines are difficult to model correctly since they are subjected to highly non-linear flow conditions under which detailed information is hard to obtain.

Another approach is to build so called nonparametric models, which basically exploit the process measurements in order to construct models of the system. These two approaches, mechanistic and nonparametric, are normally termed white-box and black-box modeling. One main advantage of black-box modeling is that the model can be developed in a cost-effective manner, and its accuracy can be validated directly against the measured data. When a model of a component is used for fault detection purposes it is fed with the same inputs as the actual component and, under normal working conditions, the outputs of the model should coincide with those of the actual system.
Recently, much attention has been devoted to improved monitoring systems for power plants. The reason is twofold: first, an increased competition on the electricity market has forced the power plant operators to optimize plant operation. This means that faults should be detected early so that correct measures can be taken, but also that false alarms should be avoided. The second reason is a growing shortage of knowledgeable personnel - a prerequisite to effectively troubleshoot and prevent failures in a power plant. With a good monitoring system, personnel dedicated to analysis is only required once a failure has been detected.

Today, a large number of parameters are measured and saved in databases to be used for historical analysis etc. However, a full analysis of this data requires extensive resources and it can be claimed that the benefits of the storage are not yet fully exploited. A data-driven, nonparametric modeling approach would provide a more effective use of the measured data and thus lead to a full utilization of the capabilities/opportunities provided by the historical records of the system behavior.

1.1 Background

An increase in efficiency as well as reliability, availability and maintainability (RAM) of a power plant is always of interest for a plant owner, especially when the competition toughens. By improving the monitoring of a power plant or its components, there is a contingency of increasing the RAM. This can be illustrated by imagining an impending failure in a gas turbine compressor. If the failure is detected at an early stage by the monitoring system the gas turbine can be shut down before a full compressor failure is developed. Consequently, the required resources for the repair, as well as the shut-down period, can be limited, thus affecting the reliability positively.

Tools that continuously monitor the performance could also allow the periods between maintenance to be extended. Currently, maintenance is performed according to schedules based on equivalent operation hours (EOH). Every stop for maintenance is very costly, as a result of production losses and expensive spare parts. By extending the periods between maintenance and
basing the stops on condition rather than on EOH, the availability can be
increased as well.

1.2 Objectives

The main objective for this research has been to develop and evaluate ANN
tools for condition monitoring and sensor validation of gas turbines. Other
more specific issues have been to:

- Develop ANN models for a variety of gas turbines and gas turbine-
based power plants.
- Identify the resources needed to develop the ANN models as well as
assessing their generalization potential.
- Identify real-life aspects associated with the development of ANN
models using operational data.
- Identify real-life aspects associated with the integration of ANN
models in a power plant’s computer system.
- Develop user-friendly graphical interfaces both for offline and online
use.
- Incorporate economic factors for operation and maintenance planning.
- Investigate whether simulation data is suitable for ANN training in
order to overcome the unavailability of operational data for new gas
turbines.

1.3 Limitations

Although there exists several kinds of artificial intelligence tools and neural
network types, only the feed-forward ANN, also known as the multi-layer
perceptron (MLP), has been utilized in this work. No effort has been made to
compare the various types of artificial intelligence since ANN is, based on
previous research at the department, deemed suitable for the high dimensional
modeling of gas turbines. Focus has instead been directed to finding and developing different applications with the selected tool.

1.4 Methods

The training of any ANN model is normally preceded by a system study where the system configuration and operational conditions are in focus. This is necessary in order to establish suitable input and output structures for the ANN model. The training process always involves the variation of one or more parameters where e.g. the number of neurons in the hidden layer is varied and the best converging network subsequently used. The number of neurons determines the complexity that can be approximated by the neural network. The nonlinear training phase is tackled in cooperation with recent development in computer capacities and by numerical training algorithms that provide efficient and fast learning/training.

A sensitivity analysis is performed to verify the possible redundancy of any input parameter in a network. It is desirable to use the simplest possible network structure with the least number of input parameters. The utilization of a simple network structure is motivated by the fact that it is less susceptible to network overfitting. The fact that it employs fewer input parameters is also motivated by the model being executable with a reduced number of available measurements.

The developed model is then utilized to validate new process measurements. Deviations from the model outputs compared to the real data, given the same input boundaries, indicate a change in the system which can be used to generate an alarm to the operator. This way, developing faults and degradation can be estimated and provide quantitative information about the actual condition of the gas turbine.
1.5 Outline of the thesis

The present thesis includes five scientific papers, preceded by the theoretical background of artificial neural networks and an overview of gas turbine monitoring. The research has been conducted at the Division of Thermal Power Engineering, Department of Energy Sciences at Lund University in close cooperation with members from the gas turbine industry and energy sector.

Chapter 1 gives a background to why the subject presented in this thesis is of interest, outlines the objectives and limitations for the studies and describes the methods used. Chapter 2 starts with a general description of artificial intelligence, followed by an explanation of artificial neural networks and leads up to the multi-layer perceptron which is the artificial intelligence approach of choice for the studies presented in the scientific papers. Chapter 3 presents various monitoring approaches for gas turbines, more specifically condition monitoring, fault detection and isolation and sensor validation. Chapter 4 gives a summary of the thesis and Chapter 5 introduces the papers included in the thesis.

1.6 Acknowledgements

The financers of this research, the Swedish Gas Centre (SGC) and the Thermal Engineering Research Institute (Värme forsk), are greatly acknowledged for their support. Corfitz Nelsson at SGC deserves special thanks due to his belief in our research.

My supervisors, Bengt Sundén and Marcus Thern, deserve a word of thanks for their guidance during this time.

From Siemens I would like to express my gratitude to the late Agne Karlsson for his support and help me in numerous matters.
2 Artificial intelligence

This chapter provides a brief introduction to the field of artificial intelligence and, more specifically, to artificial neural networks. The last part of the chapter is focused on the multi-layer perceptron, which is the network structure used in the research articles.

Artificial intelligence (AI), or computational intelligence, is intelligence ascribed to a computer system, or research aiming at constructing computer systems that demonstrate intelligent behavior. The purpose is to artificially resemble a brain’s capacity to draw conclusions, plan, solve problems, learn etc. [1].

This research area is relatively new and the term artificial intelligence was established as late as 1956 at the now famous Dartmouth Summer Research Conference on Artificial Intelligence organized by John McCarthy\(^1\) [2]. However, the dream of intelligent machines has existed ever since the antiquity. As an example can be mentioned that, 800 B.C., the famous Greek poet Homer described “tripods” as the mechanical assistants of the gods [3]. A couple of centuries later, Aristotle, the Greek philosopher, student of Plato and teacher of Alexander the great, described syllogism as a method of formal, mechanical thought [4]. These are only a few examples that have formed the culture and inspired philosophers, writers, scientists, etc. for over two

---

\(^1\) An American computer and cognitive scientist responsible for coining the term “artificial intelligence”
thousand years. More recent versions of intelligent machines (or intelligent non-humans) can be found in the works of nineteenth and twentieth century science fiction writers Jules Verne and Isaac Asimov [3]. Over the years, several disciplines have contributed ideas, viewpoints and techniques to what we today refer to as AI, in *Artificial Intelligence – A Modern Approach* [5], these areas are listed, in chronological order, as:

- Philosophy (B.C. - present)
- Mathematics (800 - present)
- Economics (1776 - present)
- Neuroscience (1861 - present)
- Psychology (1879 - present)
- Computer Engineering (1940 - present)
- Control theory and Cybernetics (1948 - present)
- Linguistics (1957 - present)

Although AI, in some form, has been on peoples’ minds for a long time, it was not until the second half of the twentieth century, much on behalf of birth of the modern digital computer, that AI development truly started. To give some examples, the first work recognized as AI, despite that the term AI was yet to be established, was presented in the *Bulletin of Mathematical Biophysics* in 1943 by McCulloh and Pitts. In their article *A Logical Calculus of the Ideas Immanent in Nervous Activity* [6], they showed that any computable function could be calculated by a network of connected neurons. They also proposed that a neural network could learn. The first complete vision of AI was presented by Alan Turing in his article *Computing Machinery and Intelligence* [7], where he posed the question “Can machines think?” and also introduced terms such as the Turing test\(^2\), machine learning\(^3\), genetic algorithms\(^4\) and

---

\(^2\) This is a proposed test to show a machine’s ability to demonstrate intelligence. The machine (or computer) passes the test if an interrogator cannot distinguish whether or not a number of written answers, to posed questions, come from a person.

\(^3\) The machine (or computer) possesses the ability to adapt to new circumstances and detect and extrapolate patterns, which is a requirement for passing the Turing test.

\(^4\) Similarly to evolution, genetic algorithms render a series of small mutations to a machine code program and preserve the ones that seem useful [8].
reinforced learning\(^5\). For a comprehensive historical review of AI, chapter 1 (Introduction) in *Artificial Intelligence – A Modern Approach* [5] is highly recommended.

Over years of development several approaches within the field of AI have emerged from which two main frameworks can be distinguished, i.e. symbolic AI and connectionist AI. The symbolic AI approach is based on logic and uses sequences of rules to instruct the computer on what to do next. Generally, a top-down strategy is applied in this approach, which means that if a problem cannot be solved in a single step it will be continuously broken down into smaller pieces. Such a system consists of many, more or less complex, IF-THEN rules, and although it is based on logic the outcome may not appear so. In this way, human expert knowledge can be built into a computer system, known as a knowledge-based system, and then used to simulate the performance of an expert. The connectionist AI approach is purely numerical and a representative example of this group is artificial neural networks (ANN), in which several units work in parallel. These units are connected to each other with adjustable links that can either magnify or limit the signal from neighboring units. Contrarily to the symbolic AI approach, the connectionist AI approach uses a bottom-up strategy starting with small elements, such as the artificial neurons, which are then interconnected in various ways in order to determine larger scale phenomena. The symbolic AI approach is suitable for performing logic and following action sequences such as conducting efficient, systematic search procedures. The major weakness arises for weak interactions, as might be the case for pattern recognition. This is, however, the strength of the connectionist AI approach which, on the other hand, performs poorly when it comes to e.g. logic. Basically, the advantages of the symbolic AI approach constitute the drawbacks of the connectionist AI approach and vice versa. This is why hybrid systems, that combine the two approaches and their different advantages, are also under development. [5, 9]

\(^5\) This is a type of machine learning where the “agent” is programmed by reward and punishment but without specifying how the task is to be achieved.
Today, artificial intelligence exists in many shapes and forms and is employed for a large variety of tasks in numerous areas, including speech and handwriting recognition, data mining, medical diagnosis, game playing, robotics and logistics planning. Other applications that many people come into contact with on a daily basis are e-mail spam filtering and Internet search engines.

The remainder of this chapter is dedicated to artificial neural networks, as this data modeling tool was used throughout the research described in the scientific papers. The choice of a suitable AI approach for a given task generally requires much trial-and-error work and the decision was in the present case taken to rely on previous research at the department, which demonstrated that ANN was a good candidate for e.g. simulation, fault diagnosis and sensor validation of heat- and power plants. [11, 12]
2.1 Artificial neural network

ANN is a non-linear statistical data modeling tool mimicking the neural structure of the human brain, and it basically learns from experience (i.e. adaptive) [13]. Instead of being built a priori from specification, neural and adaptive systems use external data to automatically set their parameters. ANN can be used to solve a variety of tasks, including classification, regression, general estimations problems, etc. An ANN consists of a group of interconnected artificial neurons processing information in parallel. The performance of a network can be improved by rendering it “aware” of its output(s) through a performance feedback loop that includes a cost function. The feedback is used to adjust the network parameters through systematic procedures called learning or training rules, in order to improve the system output with respect to the desired goal. This learning procedure is illustrated in Figure 2.2. [14]

![Figure 2.2: The design of an adaptive system.](image)

The progress of learning (or training) is an iterative process and involves modifying the strength of connections between the elements. The two main learning paradigms are:
• *Supervised learning*, in which both inputs and desired outputs are known. This means that the network can measure its predictive performance for given inputs.

• *Unsupervised learning*, in which the targets are unknown and the ANN has to find the underlying relationships within the data set by itself, and build clusters of data. [15]

Supervised learning is used for tasks of classification and regression, whereas unsupervised learning is more suitable for data clustering, compression and filtering tasks. There is a fundamental difference between clustering and classification: clustering represents the process of grouping input samples that are spatial neighbors, whereas classification involves the labeling of input samples via some external criterion. Clustering is an unsupervised process of grouping, while classification is supervised. These differences are illustrated in Figure 2.3. [8]

![Figure 2.3: The differences between clustering and classification. [8]](image-url)

There are several types of neural networks which can be categorized according to the topology of their connections. The two main groups are represented by
feed-forward networks and recurrent (feed-back) networks (see Figure 2.4). The feed-forward network is the simplest network where information only travels in a single direction, i.e. linearly from input to output. In a recurrent network, a later processing stage can propagate signals to an earlier stage. Recurrent networks can be used for e.g. dynamic or time-dependent mapping, while a feed-forward network is used for static mapping. Since the research presented in the scientific papers concerns static mapping, the focus of this chapter is on the feed-forward network. For more detailed information with regard to recurrent networks and their possibilities, *Neural and Adaptive Systems* [14] is recommended.

![Figure 2.4: The main ANN architectures.](image)

### 2.1.1 The artificial neuron

The artificial neuron is relatively simple in its construction but still represents the building block in an artificial neural network. In a true connectionist bottom-up approach, the artificial neuron should be described before going further into training algorithms and other network configuration considerations. Figure 2.5 shows a schematic of an artificial neuron along with some common transfer functions.
In the neuron, referred to in this example as “neuron k”, the incoming signals from other nodes or neurons are joined by a propagation function that is normally a standard summation. The resulting signal is called the effective input, $s_k$, and is passed through a transfer function, $F$, whereupon an output signal, designated $y_k$, is generated. The incoming signals have all been weighted, which means that they are multiplied with respective weights ($w_{ka}, w_{kb}$) equivalent to their significance for neuron $k$. Consequently, the signals are either inhibited or exhibited. Besides the inputs from the nodes and neurons, there exists an additional input to every neuron, called a bias. The bias has the value +1 and its corresponding weight is $w_{k0}$. The purpose is to present an offset to the transfer function, thus allowing the neuron to have an output even if the input is equal to zero. Equations 2-1 and 2-2 correspond to the mathematical representation of this process.

$$s_k = \sum_{j=a}^{b} (y_j \cdot w_{kj}) + w_{k0} \quad 2-1$$
\[ y_k = F(s_k) \quad 2-2 \]

2.1.2 The single-layer feed-forward network

The single-layer network is defined in the sense that there is only one layer with artificial neurons. The simplest single-layer feed-forward network is called the Perceptron and has only a single neuron. The Perceptron convergence algorithm uses supervised training in order to update the connection strengths (weights). As explained earlier, in supervised training, the desired outputs are known and employed in a cost function to calculate an error that is then used by the training algorithm to update the system. Presuming that the desired outputs of the data set are known, an error can be defined as the difference between this desired output, also known as target, and the output calculated by the ANN:

\[ e(n) = d(n) - y(n) \quad 2-3 \]

However, prior to the calculation of errors, the weights have to be assigned random starting values. When the weights have received their starting values, the first input vector, \( \mathbf{x} \) in Equation 2-4, is presented to the network, which then generates an output. This output is subsequently used to calculate the first error, employed in the perceptron training algorithm to calculate the next set of weights, i.e. the new training vector (according to Equation 2-4). When the weights have been updated, a new input vector is presented to the network and the whole procedure is repeated until the iteration receives the order to stop. The learning rate factor, \( \eta \), varies between 0 and +1.

\[ \mathbf{w}(n + 1) = \mathbf{w}(n) + \eta \cdot e(n) \cdot \mathbf{x}(n) \quad 2-4 \]

The Perceptron uses a threshold transfer function rendering it suitable for simple classification tasks. The graphical representation of a two-dimensional (i.e. two inputs) classification is illustrated in Figure 2.6, where the weights determine the slope of the line and the bias determines the distance of the line from its origin, i.e. offset.
Different learning rules basically represent different ways of updating the weights. The adaptive linear element (Adaline), which was developed subsequent to the Perceptron, uses the least mean square (LMS) learning rule (also known as the delta rule). The Adaline resembles the Perceptron in its construction, but instead of a threshold function the Adaline uses a linear transfer function. With a linear transfer function, the outputs from the ANN are not limited to only two discrete values and the Adaline can therefore be used for function approximation tasks. The idea behind the LMS learning rule is to introduce a cost or error criterion, $E$, that has to be minimized in order for the optimal solution to be found:

$$E(w) = \frac{1}{2} \cdot e^2(n)$$  \hspace{1cm} 2-5

The error, defined in Equation 2-3, can be expressed as:

$$e(n) = d(n) - x^T(n) \cdot w(n)$$  \hspace{1cm} 2-6

The main assumption is that the optimal weights should be found in the direction of the descent gradient of the error function with respect to the weights (steepest descent method):
\[ \Delta \mathbf{w}(n) = -\eta \frac{\delta E(\mathbf{w})}{\delta \mathbf{w}(n)} \] \hspace{1cm} 2-7

Differentiating Equation 2-5 gives:

\[ \frac{\delta E(\mathbf{w})}{\delta \mathbf{w}(n)} = e(n) \frac{\delta e(n)}{\delta \mathbf{w}(n)} \] \hspace{1cm} 2-8

Equation 2-6 and 2-7 give rise to:

\[ \frac{\delta e(n)}{\delta \mathbf{w}(n)} = \frac{\delta (d - x^T \cdot \mathbf{w})}{\delta \mathbf{w}(n)} = -x(n) \] \hspace{1cm} 2-9

The final result, Equation 2-10, appears identical to Equation 2-4 but is in this case not limited to adopting discrete values.

\[ \mathbf{w}(n + 1) = \mathbf{w}(n) + \Delta \mathbf{w}(n) = \mathbf{w}(n) + \eta \cdot e(n) \cdot x(n) \] \hspace{1cm} 2-10

The gradient-based learning method is illustrated in Figure 2.7.

Figure 2.7: A gradient-based updating of the weights. [10]
The Perceptron and the Adaline are two examples of simple single-layer feed-forward neural networks, but the disadvantage of single-layer networks resides in their limitation to only create linear classifiers and representing linear functions. The difference between linearly and non-linearly separable classes can be seen in Figure 2.8.

Figure 2.8: The difference between linearly and non-linearly separable classes.

This can be generalized to spaces with higher dimensionality. For instance, in a three-dimensional space, the two classes should be separated by a plane. To sum up, the single-layered topology of the Perceptron and the Adaline is capable of creating decision borders formed by a single hyperplane, and therefore, solely linearly separable problems can be solved.

2.1.3 The multi-layer feed-forward network

Multi-layered networks permit the creation of decision borders formed by several hyperplanes, and thereby, the solution of non-linearly separable tasks. Multi-layer feed-forward networks are often called multi-layer perceptrons, because of their similarity to the Perceptron [16]. Multi-layer
perceptrons (MLPs) have at least two layers of neurons, one of which is hidden. That means it only communicates within the network and not with the surroundings. In certain cases an MLP can have more than one hidden layer, but it has been proven that a single hidden layer is enough to approximate any continuous function provided that this layer has a sufficient number of units and that the transfer functions of these units are non-linear [17]. Finding the sufficient number of hidden neurons, $H$, is a trial-and-error process. The general structure of a fully connected MLP, with $M$ input nodes, $H$ hidden neurons and $N$ output neurons, is displayed in Figure 2.9.

![Diagram of a two-layered feed-forward neural network](image)

Figure 2.9: The general structure of a two-layered feed-forward neural network. [10]

MLPs are trained in a supervised manner, most frequently with the back-propagation algorithm, originally known as the generalized delta rule. The advantages of MLPs have been known for a long time but it was not until in 1986, when Rumelhart, Hinton and Williams presented the generalized delta
rule, that a satisfying way to update the weights was discovered [18]. The original delta rule was only applicable on single-layer networks with linear transfer functions. The training of a multi-layer network becomes more complicated due to the fact that the training algorithm has to affect two sets of weights, i.e. the weights between the hidden layer and the output layer as well as the weights between the input layer and the hidden layer. The backpropagation training method can be divided into two phases. First, in a manner comparable to the training of the Perceptron and the Adaline, an error is calculated and the weights between the hidden layer and the output layer are updated. Secondly, since no information regarding the desired output from the hidden layer is available, the error from the output layer is back-propagated and used to update the weights between the input layer and the hidden layer. For a detailed description of the back-propagation method, complete with equations, *Introduction of Intelligent Tools for Gas Turbine Based, Small-Scale Cogeneration* is recommended [10].

As already mentioned, the multi-layer neural network, or MLP, is suitable for non-linear classification and representation of non-linear functions. For this reason, the MLP is the primary choice when modeling the non-linear behavior of power plant systems and power plant components. The MLP’s suitability for such tasks is well established within the Division of Thermal Power Engineering and is demonstrated in e.g. *Artificial Neural Network Simulator for SOFC Performance Prediction* [19], *Artificial Neural Network Model for a Biomass-Fuelled Boiler* [20] and *Hybrid Model of an Evaporative Gas Turbine Power Plant Utilizing Physical Models and Artificial Neural Networks* [21]. The following section provides a more detailed explanation of the design- and training process of an MLP.

### 2.2 Considerations when working with MLPs

Before explaining the design process and the training of the MLP, certain information of the appropriate conditions for when to apply an MLP should be mentioned. Assuming that there is an incentive for finding a non-linear
relation between numerical data (i.e. creating a model), some basic considerations include the following:

- Since ANN is a statistical data modeling tool, a set of examples that are appropriately distributed over the input space and in sufficient numbers, must be available.
- The need for a non-linear model should be examined, since the design of a linear model is much simpler and faster. This can e.g. be done by first trying a linear model and, if it is found to be too inaccurate despite that all relevant factors are presented in the inputs, one can resort to a non-linear model.
- Finally, in cases where enough samples are available and a non-linear behavior is confirmed, one should consider whether neural networks should be employed instead of e.g. polynomials. This is a matter of how many variables the system to be modeled has, since the number of parameters in the first connection layer of a neural network increases linearly with the number of inputs, whereas it increases exponentially for polynomial approximations. Empirically, neural networks are advantageous when the number of inputs is equal to or exceeds three. [22]

2.2.1 Setup and training

Once the use of neural networks for the non-linear modeling (regression) of a system has been established, certain network specifics have to be decided on, such as these criteria:

- Number of hidden layers
- Number of neurons in hidden layers
- Transfer functions in neurons
- Error criteria
- Training algorithm
- Stop criteria
- Initial values of weights
No attempt is made to explain everything in detail. The objective is rather to provide a general overview of the decisions one might face when designing an MLP.

The **number of hidden layers** is decided based on trial-and-error. But one hidden layer is enough to approximate any continuous function, as long as the number of neurons in this hidden layer is sufficient. Additional hidden layers are seldom required for regression tasks while it might be useful for other tasks. [14]

The **number of neurons in hidden layers** is also determined through a trial-and-error process, normally reduced to find the appropriate number for a single hidden layer (explained above). In a single hidden layer network, the number of neurons in the hidden layer determines the number of hyperplanes (described in Chapter 2.1.2) portraying the function. If the number of hyperplanes (i.e. neurons in the hidden layer) is sufficient, they can form complex arbitrary shapes, matching the function. Contrarily, if they are too few, they will only follow the main trends and the representation of the function will be poor, as illustrated in Figure 2.10.

![Figure 2.10: Regression with too few hidden neurons.](image)

---

22
Too many neurons in the hidden layer will, on the other hand, result in an overfitted network with bad generalization capabilities, as demonstrated in Figure 2.11.

Figure 2.11: Regression with too many hidden neurons.

When conducting this trial-and-error process of choosing an appropriate number of hidden neurons, it is wise to start with a low number, and increase it until satisfactory results are achieved.

The **transfer function in neurons** is chosen depending on the task at hand. Two commonly used transfer functions, or activation functions, in MLPs are the hyperbolic tangent and logistic functions (see Figure 2.5). Two important aspects concerning modeling non-linear systems and using MLPs are the differentiability of the transfer function and the inclusion of non-linear parts, both of which are fulfilled by the hyperbolic tangent and logistic function.

The **error criterion**, or cost function, is a function of the error that needs to be minimized in order to improve a network’s performance. In supervised training, the error can either be defined directly by the difference between the desired output and the actual output, or by a function of the error. When training MLPs, it is common to employ the mean squared error (MSE), which is the sum of the square difference between the desired response and the actual
output. The MSE is minimized by changing the weights according to the training algorithm and, if the MSE reaches zero, the network output matches the desired outputs. [14]

The training algorithm, most commonly used together with MLPs, is the back-propagation training algorithm, also known as the generalized delta rule (see Chapter 2.1.2). The learning is based on an error criterion that is minimized with respect to the weights in the network. The challenge, as compared to the training of a single-layer network, is that no information concerning target values can be obtained from the neurons in the hidden layer. In other words, if an output unit produces an incorrect response when the network is presented with an input vector, there is no way of knowing which of the hidden units is responsible. Consequently, it is impossible to know which weight to adjust, and by how much. The solution is differentiable error criterion (or error function) and activation functions, formulated by Bishop as: “If we consider a network with differentiable activation functions, then the activations of the output units become differentiable functions of both the input variables, and the weights and biases. If we define an error function which is a differentiable function of the network outputs, then the error is itself a differentiable function of the weights. We can therefore evaluate the derivatives of the error with respect to the weights, and these derivatives can then be used to find weight values which minimize the error function, by using e.g. gradient descent. The algorithm for evaluating the derivatives of the error function is known as back-propagation since it corresponds to a propagation of errors backwards through the network.” [23]

The stop criteria can be used to avoid overtraining through the application of a cross-validation method. The cross-validation method involves measuring the network’s performance during training and, if any incentive is given, stop the training before the maximum number of iterations (epochs) is reached. This is done by splitting the data set into two parts, one training data set and one cross-validation data set. The training data set is used to calculate the errors and is thereby a part in the process of updating the weights. The cross-validation data set is not directly used in the training, but continuously verifies the networks performance with independent patterns during the training procedure. If the error (e.g. MSE) based on the cross-validation data set starts
to increase, as the error based on the training data set continues to decrease, it is an indication of overtraining.

The **initial values of weights** are normally set by randomization since no analytical solution is available. Nevertheless, certain aspects are worth considering. The initial values should be in a range so that the input to the transfer function is kept out of the saturated region. It is also of interest that the neurons learn at the same rate, which can be accomplished by providing them with equal starting conditions. [14]

For more information on any of these steps, *Neural Networks for Pattern Recognition* [23] and *Neural and Adaptive Systems* [14] are recommended.

### 2.2.2 Data management

Since data is the number one prerequisite when using a data-driven method such as ANN, the final part of Chapter 2.2 contains a list of items regarding data management:

- Filtering
- Sizing the sets
- Normalization

**Filtering** of the data is important in order to avoid that any corrupt data, such as outliers, is used in the training process. If such data is included in the training, it might have a negative effect on the overall predictive performance of the network.

**Sizing the sets** appropriately is essential in order to achieve the best results. The most important data set is the training data set where it is crucial that it contains enough well-distributed patterns. A rule of thumb is that the number of training patterns should be ten times larger than the number of weights in the network [13]. The purpose of the cross-validation data set is to test the performance of the network during training and it is normally much smaller than the training data set. The last set, the test data set, has not been mentioned until now, and is used to examine a network’s performance with
unseen data after completion of the training. A normal distribution between the data sets can e.g. be 60-25-15 (training, cross-validation and test).

**Normalization** of the data should be carried out before introducing it to the network so as to avoid that a signal with a value greater than the others becomes dominant. In order for the output signals from the MLP to be understood, they have to be de-normalized within the same boundaries. Figure 2.12 demonstrates this method.

![Figure 2.12: Pre- and post-processing of the data.](10)

Figure 2.12: Pre- and post-processing of the data. [10]
This chapter is an introduction to the research field of gas turbine monitoring, explaining some different approaches within the subareas. The subareas for this work are identified as condition monitoring, sensor validation and fault diagnosis. Condition monitoring is the process of monitoring one or several parameters of the condition in a piece of machinery, such that a significant change is indicative of a developing failure. This allows intervention in the early stages of deterioration, which is generally much more cost-effective than letting the machinery fail completely. Sensor validation is the process of monitoring sensor accuracy, detecting and isolating failing sensors and recovering sensor values. The goal is to avoid forced outages due to sensor faults. Fault diagnosis is the process of detecting and isolating faults occurring in the machinery with the same goal as with condition monitoring. In addition to early intervention, the isolation of faults acts as an aid in the maintenance process.

3.1 Background

Monitoring is a general term for which the main goal is to detect different types of performance deterioration, such as degradation, machine faults and sensor faults. Degradation and machine faults are hardware errors that indicate the health status of the gas turbine. The following conditions are typical of gas turbine deterioration [24]:

3 Gas turbine monitoring
- Fouling
- Variable inlet guide vane and variable stator vane problems
- Hot end damage
- Tip rubs
- Vibration
- Seal wear and damage
- Foreign object damage and domestic object damage
- Erosion
- Corrosion
- Control system malfunction

Any of these conditions will cause a change in the performance of a gas turbine. Fouling is a temporary deterioration that can be restored by compressor cleaning while for instance hot end damage and tip rubs require a complete overhaul of the gas turbine. To quantify the deterioration in a gas turbine, a mathematical model called baseline of the gas turbine in a healthy and new condition is used. By comparing the baseline values against the actual measured values from the gas turbine, a quantification of the current deterioration can be performed. An accurate baseline model of the gas turbine is thereby crucial for the success of health monitoring systems.

The deterioration effect will progressively worsen with increasing operating time. The rate of deterioration can be divided into three main types of failure common to all machines: time-dependent, delayed time dependent and instantaneous failures [25]. The latter will occur without any warning, thereby impossible for any monitoring system to detect. The delayed time-dependent failure is one where the deterioration is not detectable until a certain point in time. This type of failure will degrade an engine in a well-defined rate. The two time-dependent failure trends are candidates for a monitoring system.

The quality of the quantification of the current performance deterioration at the current time is dependent on accurate sensor values. Corrupted sensor values might indicate performance deterioration and may contribute to faulty decisions with regard to maintenance, which in turn will affect the reliability of the power plant negatively. Most often, hardware such as compressors and turbines are much more reliable than the sensor and control system [26].
Therefore, before using the measured data from the gas turbine, data filtering and sensor validation should be applied to it.

Engine health monitoring systems consist of several different methods, of which gas path analysis (GPA), presented by Urban in 1969 [27], is one of the most famous. GPA, as introduced by Urban, is a linear differential method where changes in the measured parameters are used to evaluate the deteriorated components. Later on, GPA was combined with the optimal state estimation method called a Kalman filter to come to terms with uncertainties in the measurements. Versions of the GPA, where statistical algorithms have been used to estimate sensor errors, are presented in for instance [28] and [29].

All gas turbines and power plants are equipped with numerous sensors measuring their performance during operation. The primary reason for this is to supply various control systems with information. Basically, control systems use incoming, measured data to correct a signal for a control device with the aim of reaching a set point for a given parameter. This ensures the correct operation of the plant or component in all operating conditions, including transient conditions. There are several “control loops” in a power plant which control everything from component-specific parameters through process parameters to main plant parameters. Besides being used for the control of the power plant, the stream of data is employed to monitor the equipment and generate warnings, alarms and even induce complete shut-downs of the operation if needed. The development of data-driven monitoring tools, such as ANN-based condition monitoring and sensor validation, is made possible much on account of this generated operational data.

3.2 Condition monitoring

The main goal of condition monitoring is to optimize the maintenance cost of a power plant. The three principle maintenance strategies include:

- Run-to-failure: the plant is basically operated until it breaks down.
- Preventive: a planned maintenance service schedule is applied.
• Predictive (condition-based maintenance): the maintenance schedule is planned according to the actual condition of the plant or component.

The run-to-failure method has several drawbacks. First, a failing component may cause failure in other components, which can lead to the total breakdown of the gas turbine. Another disadvantage is that the failing component might cause a shut-down during a critical time where the power is needed the most. Preventive maintenance has been the main method for a long time and is based on regular intervals. However, this method is also accompanied by certain drawbacks. For instance, failures do not occur regularly and might take place unexpectedly in between the overhauls. Also, this method will cause unnecessary overhauls of the components, resulting in time-consuming maintenance and major profit losses for the power plant owners. Predictive maintenance on the other hand is based on the actual condition, hence the name condition monitoring, and maintenance can be planned at an optimal time, which would generate the highest profit for the power plant owners.

Condition monitoring addresses additional challenges, since it requires a health monitoring system which, at any point, can convey the health status of the plant. Practically, this is done by comparing the actual performance against a mathematical model of the gas turbine representing the performance in a healthy status. Differences between the actual performance and the baseline model can be interpreted as performance deterioration. The “health” of a gas turbine is normally expressed as the efficiency of the components; parameters that cannot be measured directly. Instead, the health factors have to be calculated through measurable values, e.g. temperatures and pressures along the gas path of the engine.

GPA addresses this problem by relating the changes in the measured parameters to changes in the health parameters, and since GPA is a differential method, changes in the health parameters are monitored rather than absolute values. The literature on GPA approaches for monitoring is extensive and the following references represent the pioneering work [25, 27, 30].

Another monitoring approach consists in constructing a purely mathematical model, based on physical laws such as heat and mass balances together with
the characteristics of the components. These models can be made very accurate. However, they require sensitive information about all components which most often is highly confidential. The original equipment manufacturers (OEMs) use these models in the design phase but are reluctant to supplying them to the operators due to the sensitive information contained within the models. As another drawback, the method generally requires more input than is normally available for an “on-site” engine. Furthermore, the effort to build such a model is normally too costly for it to be considered a practical solution.

Data-driven models have emerged in the engineering community during the last 10-20 years [31]. In a data-driven model, the baseline model is developed directly from the measured data without the addition of any physical principles. Data corresponding to a healthy operation is used to train the data-driven model. There are several methods to choose between, such as radial basis function networks, feed-forward multi-layer perceptrons, polynomials etc. However, for high-dimensional systems (with more than three input parameters), the multi-layer perceptron method is considered the most attractive. This is the only data-driven method that scales linearly with the dimension, making it highly appropriate for high-dimensional modeling.

Since the 1980s, a mathematical theory for feed-forward multi-layer perceptrons has existed, which shows that the MLP is a universal approximator [17, 32]. However, the building of the model is a somewhat complex problem and today’s research is still on trial even though some guidelines are available. Much of the current research work is focused on demonstrating different applications and methodologies to apply the MLP. Data-driven modeling offers a cost-effective approach to condition monitoring, since all that is needed to build the monitoring model is data from the gas turbine.

3.3 Fault detection and isolation

Fault detection and isolation is the process of detecting a faulty data pattern, and then isolating it as a specific fault. This is not an easy task in a gas turbine, due to high intercorrelation between the components. The control as well as
the specific type of gas turbine has to be considered. The behavior of a gas turbine when a component is failing depends on its configuration. For example, a single-shaft gas turbine will not react in the same way as a two-shaft engine. Also, the current control strategy affects the failing behavior. Certain gas turbines are controlled by a constant turbine inlet temperature (TIT), while other types might have a constant power as the objective function. When a fault is quantified as present and an action is required, the isolation process plays a very important role. A correct isolation of a fault can speed up the delivery of spare parts from the OEM. For operation considerations, it is also important to isolate the detected faults. Some faults may just trigger a maintenance action, while others might require an immediate shut-down in order to prevent more serious damage to the gas turbine.

The state-of-the-art today is to use vibration monitoring as an additional tool for health monitoring [33]. Some faults cannot be detected through thermodynamic relationships, as exemplified by combustor cracks. Wear debris monitoring is therefore another monitoring technique that has been used for a long time [34]. It involves oil analysis, where an oil sample is taken and analyzed in order to detect failing components, which is applied to gears and bearings on the gas turbine.

GPA is one of the main tools used in gas turbine fault detection and isolation. However, GPA was mainly developed for aero derivatives, which are most often simpler than industrial gas turbines. The faults that are implemented in GPA are mathematical artifacts, and GPA does not take into account that a fault may cause different fault patterns depending on the operational conditions. Moreover, the aero derivative engines have fewer sensors than an industrial gas turbine due to different physical constraints. Therefore, GPA is not as suitable for industrial engines as it is for aero derivatives.

Several types of neural networks can be employed in fault diagnosis where the measured (thermodynamic properties) parameters are utilized to determine the actual faults. MLPs [35], Bayesian networks [36] and fuzzy logic approaches [37] have all been reported. According to Joly et al. [38], different combinations of neural networks can be used for the fault diagnosis task. In most cases, the neural network is utilized as a pattern recognition tool to
identify the failure pattern. Since several fault scenarios can be considered and real failure data is rarely existing and difficult to obtain, these neural network implementations are most often based on simulation data. In such cases, a mathematical model is used to simulate the faulty data which is used to train the neural network. The other option is that the neural network is trained directly on the real failure data.

The approach described by Joly et al. [38] illustrates the complex problem of fault diagnosis, especially when several faults are present. In this case, varying networks are used for different fault scenarios in order to make it easier for each network to perform its specific task. One network is trained to identify a faulty pattern where several others are trained to diagnose single and/or multiple faults. This way, a more accurate fault diagnosis can be performed.

3.4 Sensor validation

A gas turbine is monitored through measured values along the gas path. The measured values need to be validated before being used to interpret the health status of the gas turbine. Normally, the reliability of the gas turbine itself is much higher than the reliability of the sensors. Thus, validating the measured values is of great importance in order to avoid maintenance decisions based on faulty information.

Most often, there is a redundancy in the measured plant data which can be used to detect the failing sensor. However, several ways of dealing with this uncertainty exist. For example, one way is to compare the sensor values against a physical equation of the component, e.g. the energy equation. By using a weighted average, which takes into account each sensor’s accuracy, it is possible to calculate which sensor combinations that satisfy the constraint and thereby detect the failing sensor. This is explained in detail by Rodney [39].

GPA, mentioned in the previous sections as a candidate for condition monitoring, fault detection and isolation, can also be employed for sensor validation. In these cases, GPA is utilized in conjunction with statistical algorithms to estimate the sensor error, see for example the work of
Lunderstaedt et al. [28] and Doel et al. [29]. The weighted least square techniques applied to the GPA have been shown in [29, 40, 41].

Neural networks provide another approach to sensor validation, which has been extensively studied. Several types of neural networks have been applied to sensor validation of power plants in numerous ways. Sensor validation has been demonstrated through the Kohonen Map, the feed-forward MLP and the auto-associative neural network [42, 43]. Commonly, the neural network is trained with a data set either generated by a simulation program, as shown by Ogaji et al. [43], or by operational data from the system (gas turbine). It is possible to make a distinction here, between a case where the neural network is developed with simulated data and one where it is developed with real operational data. For the former, the neural network is used since it provides a simpler way of implementation for real-time monitoring than the simulation program itself. For the latter, where the neural network is developed from real data, the situation is rather different. Most often, there is no mathematical model available; only measured data from the system (gas turbine). The network is then used to build the model directly from this data.

The most common implementation of a neural network is through the auto-associative neural network (AANN), a so-called identity mapper, where the network is trained to reproduce the input values to the outputs. When a faulty sensor is present in the input data, the network predicts a more accurate value since the correlations between the parameters are learned by the network in the learning session. The network output is therefore a more reliable value than the input value itself. An AANN can mainly be configured in two different ways, i.e. with one or three hidden layers. When an AANN is configured with one hidden layer, the correlations between the parameters are learned during the training session. When three hidden layers are used, a certain compression is also applied by a so-called bottle-neck layer which consists of fewer neurons than the input/output layers. Two of the first publications to have introduced neural networks for sensor validation are reported by Kramer [44, 45]. Since then applications have been numerous [26, 46, 47].
The building of mathematical models through physical laws is a complicated process, especially for highly interconnected systems with losses (turbulence etc.) which are difficult to model in an accurate way. In this aspect, neural networks provide the option to build the monitoring models directly from the operational data, thus presenting a cost-effective option for sensor validation. Neural networks provide the capability of using high dimensional and nonlinear data in an efficient manner, characteristics which are always hard to simulate with analytical models. Another aspect worth mentioning is that sensor accuracy etc. is directly included in the neural network. This results in robust models which are insensitive to small input errors that are common in real measured data.
Data-driven modeling has emerged during the last decades as an alternative to physical simulation for monitoring purposes. One of the main reasons is the sophisticated data acquisition systems introduced during the last years, which are used to store measured data from power plants. The data itself is not worth much if it is not analyzed. However, plant operators do not have access to the detailed component characteristics needed for accurate physical modeling. Thus, using the saved data provides an efficient and cost-effective approach for the modeling.

The main goal of this thesis has been to evaluate ANN as a data-driven modeling approach aimed at fault detection, degradation estimation and sensor validation of gas turbines. By assessing several gas turbines with varying characteristics in terms of off-design behavior, the generic capabilities of the approach were validated. Another goal was to determine the economical impact caused by degradation in gas turbines.

A new technology is always met with a certain amount of skepticism, until it is proven reliable and trustworthy. In this case, the users of the models are the plant operators which normally have only limited knowledge of the gas turbine. A graphical user interface was developed and presented to the plant operators, demonstrating the usability of the ANN model. In practice, the user interface showed that the ANN monitoring model could be employed without any knowledge from the user side, which is an advantage for practical implementation. Moreover, the user interface could be operated both offline for training as well as online for real-time monitoring. Online monitoring by
the ANN models demonstrated that, as long as the ANN model predictions and real measurements were similar, the operation continued without faults. If any deviations between the model and the measurements were observed, an investigation of the underlying reason was needed. Advantageously, no effort was required until the detection of a faulty behavior. In this way, unnecessary data analysis could be avoided.

The deregulation of the electricity market has resulted in new challenges for the power producers. Combined power plants, which previously were only operated in base load, nowadays function more often in part load, caused by for instance the introduction of wind power to the market. More sophisticated monitoring systems are required, with the possibility of being used in real time to detect failures and abnormal behavior of the gas turbine as early as possible.

This thesis has shown the feasibility of introducing ANNs as a candidate for data driven modeling for gas turbine monitoring. It was demonstrated that accurate monitoring models could be developed by using measured data, without detailed information about the gas turbine. This renders it possible for plant owners to develop their own monitoring systems for their specific plants.
5 Summary of Papers


This paper describes the investigation of MLP capacities for gas turbine modeling using operational data from a Siemens SGT-600 machine. Operational data was primarily chosen to evaluate the possibilities of creating an accurate performance-predicting ANN model representing an actual machine. Besides creating a precise model, various real-life aspects, associated with the modeling of an actual machine, were examined. These real-life aspects included everything from data acquisition and filtering to specific operational conditions, such as anti-icing operation.

This study demonstrates that the use of ANN rendered it possible to create tailor-made models, with very high prediction accuracies, using operational data for specific gas turbines. Together with the developed user-friendly graphical user interface, the model can be employed for offline performance simulation for operation planning or as a tool in the training of new operators. The model also shows potential to be integrated in online condition monitoring.

The author of this thesis carried out everything from the data acquisition and filtering through the training and evaluation of the ANN models to the development of the user interface. Mohsen Assadi was the coordinator of the work and participated in the analysis of the results. Sudipta De helped compile the material into a scientific paper.
Paper II. Application of artificial neural network to the condition monitoring and diagnosis of a combined heat and power plant was presented and published at the ECOS conference in Krakow, Poland, in the summer of 2008. Upon recommendation, this paper was later also published in the Journal of Energy, Vol. 35, pp. 1114-1120, 2010.

This paper presents the ANN modeling of several power plant components using operational data. The components included a gas turbine, a heat recovery steam generator, a biomass-fueled boiler and a steam turbine. The possibility of integrating these ANN models in the power plant’s computer system and developing a graphical user interface for realizing online condition monitoring was also investigated.

This study demonstrates that ANN can be advantageously employed for modeling of several different components with high accuracy. Thanks to the limited size of the ANN models, their high computational speed and the transferability into any programming language, the integration and functionality in combination with the power plant’s computer system was successful. Together with the developed graphical user interface, condition monitoring was demonstrated to be an achievable goal.

The author of this thesis carried out everything from the data acquisition to the implementation of the ANN models in the power plant’s computer system as well as the evaluation of the complete online system. Thomas Palmé contributed with fruitful discussions during the course of the investigation.

Paper III. Condition-based maintenance of gas turbines using simulation data and artificial neural network: A demonstration of feasibility was presented and published at AMSE Turbo Expo, Berlin, Germany, in the summer of 2008.

This paper investigates the possibilities of using simulation data, instead of operational data, for ANN modeling of gas turbines in order to overcome the unavailability of data for new gas turbines. This is an attempt at enabling the delivery of an ANN tool, for condition monitoring, together with new gas turbines. Simulation data was produced with Siemens’ own design tool
representing their SGT-800 gas turbine. An ANN model was trained with this data and its performance was compared to that of an ANN model trained with operational data.

This study presented limitations since the ANN model that was trained with simulation data was not developed for this specific purpose. However, when eliminating these factors, the congruence between the ANN model trained with simulation data and the one trained with operational data was close to perfect.

The author of this thesis carried out the data acquisition and filtering, the training of the ANN model based on operational data and the comparison of the performances between the two models. The ANN model trained with simulation data was developed by Jaime Arriagada, a former Ph.D. student at the Division of Thermal Power Engineering. Mohsen Assadi was the coordinator of the work and participated in the analysis of the results. Sudipta De helped compile the material into a scientific paper.

**Paper IV.** *A novel approach for gas turbine condition monitoring combining CUSUM technique and artificial neural network* was presented and published at AMSE Turbo Expo, Orlando, Florida, USA, in the summer of 2009.

This paper reports on the combination of an artificial neural network and a sequential analysis technique for monitoring of a gas turbine. The performed work was an attempt at eliminating the need for retraining or calibration of the performance-predicting model (in this case the ANN model) as the gas turbine was deteriorated. Another benefit may also be the detection of very small gas turbine anomalies.

In this study, operational data from a Siemens SGT-600 gas turbine was employed for the training of an ANN model, which was subsequently used for the prediction of performance parameters of the gas turbine. Simulated anomalies were introduced on two sets of operational data, acquired one year apart, whereupon this data was compared to corresponding ANN predictions.
The cumulative sum (CUSUM) technique was utilized to improve and facilitate the detection of such anomalies in the gas turbine’s performance.

The author of this thesis performed the ANN model training, implementation of the CUSUM algorithm and the analysis of the results. Thomas Palmé carried out the literature review and helped in the writing of the paper. He also participated in the analysis of the results. Magnus Genrup initiated the study.

**Paper V.** *Gas Turbine Sensor Validation through Classification with Artificial Neural Networks* was presented and published at the ECOS conference in Foz do Iguaçu, Paraná, Brazil, in the summer of 2009.

This paper presents a method for evaluating sensor accuracy, with the aim of minimizing the need for calibration and at the same time avoiding shut-downs due to sensor faults etc.

The proposed method was based on the training of artificial neural networks as classifiers to recognize sensor drifts. The method was evaluated on two types of gas turbines, i.e. one single-shaft and one twin-shaft machine. The results demonstrated that the method was capable of early detection of sensor drifts for both machine types as well as of an accurate production of soft measurements.

The technical work was equally divided between the author of this thesis and Thomas Palmé, while the former was responsible for the writing of the paper. Agne Karlsson contributed with gas turbine-specific information during the course of the work.
5.1 Papers outside the thesis


6 Bibliography


[9] Minsky ML. Logical versus analogical or symbolic versus connectionist or neat versus scruffy. AI MAGAZINE. 1991;12:34.


Development and multi-utility of an ANN model for an industrial gas turbine
Fast M, Assadi M, De S.
Development and multi-utility of an ANN model for an industrial gas turbine

M. Fast\textsuperscript{a,\*}, M. Assadi\textsuperscript{a}, S. De\textsuperscript{b}

\textsuperscript{a}Department of Energy Sciences, Lund University, P.O. Box 118, SE 22100, Sweden
\textsuperscript{b}Department of Mechanical Engineering, Jadavpur University, Kolkata 700032, India

Abstract

Demonstration of different utilities for industrial use of an artificial neural network (ANN) model for a gas turbine has been reported in this paper. The ANN model was constructed with the multi-layer feed-forward network type and trained with operational data using back-propagation. The results showed that operational and performance parameters of the gas turbine, including identification of anti-icing mode, can be predicted with good accuracy for varying local ambient conditions. Different possible applications of this ANN model were also demonstrated. These include instantaneous gas turbine performance estimation through a graphical user interface and extrapolation beyond the range of training data.

1. Introduction

The power and combined heat- and power (CHP) sectors all over the world have gone through major transformations over the last two decades due to rising competitiveness for deregulation of electricity and more stringent laws for environmental protection. Increasing the reliability, availability and maintainability of existing plants is a present need to improve the performance and minimize the environmental impact. Conventional simulation and performance monitoring as well as maintenance schedules need significant improvement to cope with this present need. Advanced simulation tools and condition monitoring systems are thus very crucial for modern plants. Some simulation and condition monitoring tools use heat- and mass balance programs \cite{1}, which often rely on physical and thermodynamic laws. These are very powerful regarding thermodynamic studies and the estimation of parameters (efficiency, power output etc.). However, they require significant expertise and man-hours for monitoring the condition of the plant and locating faults. Several artificial intelligence (AI) based tools are available \cite{2,3}, ANN is one such AI based tool suitable for e.g. simulation and condition monitoring of power and CHP plants. ANN is a computer algorithm that can learn patterns through proper training, i.e. adaptive \cite{4}. Because of this feature, these are often well suited for modelling complex and non-linear processes of real life. Feasibility of ANN applications at the process/component level \cite{5–7} as well as at the plant/system level \cite{8–10} has been reported by several authors. These are considered to be modern ‘high value, low cost’ IT-based ‘intelligent’ tools to substitute for conventional simulation and condition monitoring tools. Thus ANN has been demonstrated to be a useful technique for accurate and realistic modelling of real life plants aiming at better prediction of performance and environmental impacts.

Researchers at the division of Thermal Power Engineering at Lund University has carried out several investigations \cite{1,11–21} in this field and ANN has shown to be a good candidate for fault diagnosis, process identification and modelling of non-linear systems in the energy field and this work is a continuing study of those previous studies. In this work a demonstration project has been reported in collaboration with Lunds Energi, Lund, Sweden. Operational data from a gas turbine based cogeneration unit at Gunnesbo, Lund, Sweden has been supplied by Lunds Energi. Researchers at the division of Thermal Power Engineering were assigned with the task of developing an ANN model with this data to demonstrate the multi-utility of such a model for real life plants. The novelty of this study lies in that a complete cycle, from data acquisition, data screening, ANN training and evaluation of developed ANN model, development of a graphical user interface and delivery of a finished product to the plant, has been completed.

Real life aspects, such as anti-icing operation, have also been taken into account. In previous studies simulation data was often used to establish the use of ANN for modelling of power plant systems rather than focusing on delivering an utilizable product.
When the training is completed the weights are fixed and the net-squared error (MSE) is calculated between each epoch (iteration) of the network, layer by layer, and a set of outputs is obtained. Errors are then used for updating the weights before another set of inputs are transmitted through the network. The data used as inputs is transmitted through the network, multiplied by their adjustable weights and in each processing element a continuous function is generated by comparing these outputs with the desired outputs through an iterative process called training. Developed model can be used for continuous condition monitoring of the plant by comparing the actual sensor data during operation of the plant with the predicted data from the model. Moreover, all the input data (local ambient conditions) may be assumed without operating the plant and an estimation of expected performance of the plant can be done before actually starting, which means that the model is even useful for offline applications. The plant has to be operated with anti-icing measures during simultaneous cold and humid ambient conditions to avoid the very harmful possibility of ice formation on the aerofoil blades in the compressor. Switching over to anti-icing and coming back to normal mode of operation has also been modelled with ANN and integrated to develop a unified ANN system for all possible modes of operation. Finally an easy-to-use graphical user interface has been developed so that the model can be used by any person for monitoring, simulation or learning of the plant basics.

2. Brief basics of ANN

ANN is a simulation tool that mimics the neural structure of the human brain [4]. The brain basically learns from experience. In contrary to traditional mathematical models, which are programmed, ANN learns the relations between selected inputs and outputs through an iterative process called training. ANN consists of a number of interconnected artificial neurons with linear or non-linear transfer functions and is well capable of predicting non-linear behaviour of a system. The multi-layer feed-forward network is the type of network which has been used during this study. It consists of an input layer, one or more hidden layer(s) and an output layer. However, there is no impediment to having more than one hidden layer, since it has been proved that one layer with hidden neurons is enough to approximate any continuous function if it only has a sufficient number of neurons [22].

Once the inputs are presented to the network they will be multiplied by their adjustable weights and in each processing element summed and passed through a transfer function in order to produce outputs. The data used as inputs is transmitted through the network, layer by layer, and a set of outputs is obtained. Errors are generated by comparing these outputs with the desired outputs. The errors are then used for updating the weights before another set of inputs are transmitted through the network. The mean squared error (MSE) is calculated between each epoch (iteration) and the training is terminated when the MSE is satisfactory low. When the training is completed the weights are fixed and the network is ready to predict outputs from previously unseen data, i.e. generalization.

3. Brief description of the gas turbine and CHP plant

The schematics and main operating parameters of the CHP plant, with its gas turbine and heat recovery unit for the district heating system, is shown with a screenshot from an operator station in Fig. 1. The gas turbine was installed in 1991 by ABB STAL, now Siemens Industrial Turbomachinery. It was originally a GT10A but was rebuilt to GT10B in that same year. The GT10 (now known as SGT600) series gas turbines are lightweight industrial gas turbines. These are designed and developed to incorporate size and weight advantages of the aircraft derivative gas turbine while at the same time maintaining the robustness, flexibility and long life advantages of the traditional industrial gas turbines. The compressor is of axial flow type, it has ten stages with an overall pressure ratio of 13:6:1. The combustion chamber is of annular type and is built for dual fuels, i.e. gas and liquid. However, Lunds Energi presently uses natural gas as fuel. It has in total eighteen dual fuel air atomized burners. The turbine consists of a power turbine and a compressor turbine. The compressor turbine has two stages with internal cooling by convection. It has a nominal speed of 9770 rpm with a rated air flow of 77.2 kg/s. The power turbine is a heavy duty industrial turbine with a nominal speed for power generation of 7700 rpm. The gas turbine has the capacity of producing 22 MW of electricity [23,24].

The hot exhaust from the gas turbine is passed through the heat recovery unit to produce hot water for the district heating system. Heat is extracted in an economizer and an exhaust gas condenser. A regulator continuously adjusts the valves located on the district heating water pipe which automatically controls the water flow rate. A constant gas temperature, at the exit of the exhaust gas condenser, and a desired district heating water delivery temperature are thus maintained. Approximately 35 MW of heat is extracted in the heat recovery unit [24].

4. ANN model development

The overall objective of this work was to demonstrate the practical usefulness of ANN modelling for an existing gas turbine in a cogeneration plant. Scopes for using ANN for multi-utility objectives for a real life plant with modest cost and effort were demonstrated by this work. For the experienced ANN modeller approximately one month is required for data collection and filtering, system studies and training of the neural networks. Operational data was obtained from the plant owner and used for the ANN model development. However, as the accuracy of prediction of a trained ANN can never be better than that of the training data set, proper screening of obtained data is very important. Moreover,
proper selection of input and output parameters is also of special significance. Finally, the purpose of the developed model also influences the selection. The best possible ANN as well as its training was obtained by system knowledge, ANN experience and trial and error. The trial and error procedure includes a sensitivity analysis to check the redundancy of any input parameters. These steps, for the development of the gas turbine ANN model, are described in subsequent chapters.

4.1. Data collection and screening

The gas turbine of this plant is always set to run at full load. However, it is only economically beneficial to operate the gas turbine during the winter months due to local conditions, such as ambient temperature, district heating demand, electricity and gas prices. Operational data of one week duration from the plant was obtained as averaged for one minute. Thus this available data was only suitable for modelling the steady state operation of the plant. The data obtained was considered to be the baseline data for definition of the healthy condition of the plant. Obtained data was screened to remove any transient data during start or closure of turbine operation. Otherwise these transient data would confuse the ANN during training and reduce the accuracy of prediction of the final model for the steady state operation. Also this plant is equipped with anti-icing provisions. When required the anti-icing measure is activated to prevent formation of ice in the intake manifold during high humidity and simultaneous cold ambient conditions. The switching between anti-icing and normal mode of operation also causes transient operation of the plant. These transient data were also removed from the training data set. However, it is also to be remembered that all the operational data will change even at steady state operation, due to variations in ambient conditions. Furthermore, all corrupt data, e.g. obvious erroneous sensor signals, were removed before training the ANN. In total approximately 4000 data points were removed from the original data set. After the removal of unsuitable data 7500 data points remained and were hence used in the training and validation process.

4.2. Selection of input and output parameters

Proper selection of input parameters for accurate prediction of a set of output parameters is important for ANN modelling of any system. System knowledge to identify the interrelation between the input and output parameters helps in this respect. However, availability of reliable measured plant data for any parameter is also a precondition for inclusion of that parameter in the ANN model. Since the gas turbine is always set to run at full load the operational performance variation of the plant is mostly due to the change in local ambient conditions and fuel quality. The two most desired outputs from the model were electric power and fuel flow. Prediction of variation of these two with the change of ambient conditions was decided and hence these are included as output parameters. Apart from these two quantities, several other quantities are also included as output parameters. These are selected based on real life need as well as availability of their reliable measured value from the plant. The complete sets of input and output parameters are included in Table 1. Gas flow was estimated in unit MJ/s due to the fact that the mass flow of gas (kg/s) was multiplied by a constant calorific value (MJ/kg). Relevance of each of the input parameters, for the prediction accuracy of the selected output parameters, has been examined by sensitivity analysis (chapter 4.4). Input parameters for which the accuracy of prediction of output parameters did not improve was considered redundant and removed from the final ANN model. One important issue related to this plant is the anti-icing. For certain low temperature and high
humidity condition of the local ambient, ice formation starts in the intake manifold and to avoid this anti-icing is activated by bleeding some amount of air from the compressor delivery to the intake manifold through a regulating valve. Thus during operation of the gas turbine in anti-icing mode, the performance decreases from that of normal mode. For the simplicity of using this developed model the switching between normal and anti-icing mode of operation has been implemented by training an ‘anti-icing ANN’ with the available data. Thus the unified ANN system initially decides the mode of operation on the basis of local ambient conditions and then predicts the values on the basis of that decision. Thus local ambient conditions were not only used as input parameters for the prediction of output parameters but also for the initial selection of the mode of operation of the gas turbine. The development of an ANN for deciding regarding anti-icing or normal mode of operation has been described in the next subsection. This acted as an initial ‘switch’ for deciding the possible mode of operation of the plant based on local ambient conditions.

4.2.1. Anti-icing

One initial problem encountered was that when the anti-icing system was in operation the gas turbine’s performance would differ significantly. Training an ANN with measured data from both when the anti-icing system was in operation and when it was turned off can be compared with training an ANN with data from two different plants. The obvious solution to this problem was to separate these data sets and train two different ANNs. However, a better and more sophisticated way to tackle the problem was to add an extra input parameter with the value ‘1’ if the anti-icing system was in operation and the value ‘0’ otherwise. This assisted the ANN to differentiate between these two modes of operation. Since data describing the mode of operation (‘1’ or ‘0’) did not exist, it was added to the original data set manually in Excel. The principle of deciding regarding the switching on or off of the anti-icing was to study the compressor inlet temperature. When there was a large change in this quantity, within a small timeframe, it indicated that the anti-icing system was turned on or off. Table 2 demonstrates a sample data set for such condition. The transition was not instantaneous but actually more gradual than the table shows. This is because data representing the transition period was not used for the training of the ANN and the period from when the anti-icing system was turned off or on until the gas turbine was stabilized in its new mode of operation was removed.

4.3. ANN training parameters

Training of the ANNs was done using the commercial software NeuroSolutions. Back-propagation was the basis of training for this supervised neural network. The data used as inputs were transmitted through the network, layer by layer, and a set of output data were obtained. Before this forward pass the weights of the network received randomized values. The obtained outputs were compared with the desired output values and, as a backward pass; the difference between desired outputs and calculated outputs (error) was used to adjust the weights of the network in order to reduce the level of the error. This process is called ‘supervised learning’. This is an iterative process, which continues until an acceptable level of errors is obtained. Each time the network processes the whole set of data (both a forward and a backward pass), is called an epoch. The network was trained in this way and the error was reduced by every epoch until an acceptable level of error was obtained. This method is called error back-propagation training [15]. According to the previous experience [11–21], non-linear transfer functions produce good results for the modelling of real life plants. For this work the tangent hyperbolic transfer function was found to be the most suitable transfer function and was used for the modelling. With some trials the final configurations for the best possible modelling of the plant were optimized for 10,000 epochs, a variation with 1–10 hidden neurons (H) with 3 runs for each configuration where the best converging network was saved. The data was, based on experience, normalized between −0.8 and 0.8 which gives the trained network good extrapolation capabilities while at the same time the non-linear characteristics of the tangent hyperbolic transfer function is not lost. Training with a larger variation, e.g. 11–20 H, did not improve the results. Also, the cross-validation method was used, i.e. the performance of the network was checked during training. This enabled NeuroSolutions to stop the training before the maximum number of epochs was reached, which avoids overtraining of the neural network. The performance was checked with the cross-validation data set and when for 200 epochs the MSE did not decrease the training stopped automatically. The filtered plant data was divided as 60% for training, 15% for cross-validation and rest 25% as test data set.

4.4. Sensitivity analysis

The purpose of a sensitivity analysis is to determine the dependence of output parameters on the initially selected input parameters. The input parameters are initially selected based on system knowledge and availability of reliable data. However, ANN being only a data driven method, no physical relations of processes are implemented in this modelling. Thus interrelations between input and output parameters are not imposed by any equation but this is implicitly implemented during the training with the data. However, to ascertain the definite dependence between selected input and output parameters, the effects on the prediction accuracy of the output parameters by the trained ANN has to be compared to the prediction accuracy of ANN models without each of these input parameters. If the prediction accuracy is not affected in spite of removing an input parameter, it proves that this input parameter is redundant and may be removed from the input set of parameters.

---

**Table 1**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Relative humidity</td>
<td>%</td>
</tr>
<tr>
<td>Ambient pressure</td>
<td>bar</td>
</tr>
<tr>
<td>Ambient temperature</td>
<td>ºC</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Mass flow rate of air</td>
<td>kg/s</td>
</tr>
<tr>
<td>Compressor outlet pressure</td>
<td>bar</td>
</tr>
<tr>
<td>Compressor outlet temperature</td>
<td>ºC</td>
</tr>
<tr>
<td>Mass flow rate of natural gas</td>
<td>MJ/s</td>
</tr>
<tr>
<td>Turbine outlet temperature</td>
<td>ºC</td>
</tr>
<tr>
<td>Power output</td>
<td>MW</td>
</tr>
<tr>
<td>Carbon dioxide emissions</td>
<td>%</td>
</tr>
<tr>
<td>Generated heat</td>
<td>MW</td>
</tr>
</tbody>
</table>

**Table 2**

<table>
<thead>
<tr>
<th>Data point</th>
<th>Date/time</th>
<th>Relative humidity (%)</th>
<th>Ambient pressure (mbar)</th>
<th>Ambient temp. (ºC)</th>
<th>Comp. inlet temp. (ºC)</th>
<th>Anti-icing (1-on, 0-off)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7082</td>
<td>2005-02-16 02:51:13</td>
<td>79.7</td>
<td>995.5</td>
<td>-3.8</td>
<td>-3.1</td>
<td>0</td>
</tr>
<tr>
<td>7083</td>
<td>2005-02-16 02:52:13</td>
<td>79.8</td>
<td>995.5</td>
<td>-3.8</td>
<td>-3.1</td>
<td>0</td>
</tr>
<tr>
<td>7084</td>
<td>2005-02-16 02:53:13</td>
<td>79.8</td>
<td>995.8</td>
<td>-3.8</td>
<td>-3.1</td>
<td>0</td>
</tr>
<tr>
<td>7085</td>
<td>2005-02-16 02:54:13</td>
<td>79.8</td>
<td>995.6</td>
<td>-3.8</td>
<td>-3.1</td>
<td>0</td>
</tr>
<tr>
<td>7086</td>
<td>2005-02-16 02:55:13</td>
<td>79.8</td>
<td>995.8</td>
<td>-3.8</td>
<td>-3.1</td>
<td>0</td>
</tr>
<tr>
<td>7131</td>
<td>2005-02-16 03:39:13</td>
<td>80.2</td>
<td>995.8</td>
<td>-3.4</td>
<td>8.0</td>
<td>1</td>
</tr>
<tr>
<td>7132</td>
<td>2005-02-16 03:40:13</td>
<td>80.2</td>
<td>995.8</td>
<td>-3.4</td>
<td>8.2</td>
<td>1</td>
</tr>
<tr>
<td>7133</td>
<td>2005-02-16 03:41:13</td>
<td>80.2</td>
<td>995.7</td>
<td>-3.3</td>
<td>8.2</td>
<td>1</td>
</tr>
<tr>
<td>7134</td>
<td>2005-02-16 03:42:13</td>
<td>80.3</td>
<td>996.0</td>
<td>-3.3</td>
<td>8.2</td>
<td>1</td>
</tr>
<tr>
<td>7135</td>
<td>2005-02-16 03:43:13</td>
<td>80.2</td>
<td>996.0</td>
<td>-3.3</td>
<td>8.1</td>
<td>1</td>
</tr>
</tbody>
</table>
in the final model. On the other hand if prediction accuracy is reduced in the same process, the interdependence between the input and output parameters is confirmed.

In this work local ambient conditions, specified by pressure, temperature and relative humidity, were considered as input parameters for the prediction of performance of the gas turbine (refer to Table 1). The effects of each of these three input parameters on the prediction accuracy of each of the output parameters were examined. Each of these input parameters were removed one at a time keeping the other two to check the prediction accuracy. For each case a new ANN model was trained and the prediction accuracy of the output parameters from these models was compared with the original model. Moreover, the justification of considering anti-icing separately by training with available data for it has also been explored. The results of this sensitivity analysis have been presented in two steps. At first the effect of not considering anti-icing separately has been discussed followed by the discussion on the effects of each of these input parameters on the selected output parameters.

Fig. 2 shows how the prediction of power output was affected if the anti-icing and the normal mode of operations were not considered separately. From this figure it was obvious that if the ANN was trained with data from both of these two modes of operations without the information that these were two completely different modes of operation, then the obtained prediction accuracy for power (and for other parameters too) was not acceptable. The sensitivity of each of the output parameters as anti-icing switch was not considered in the ANN model is shown in Table 3.1 Thus the justification of providing information to the ANN initially to decide regarding the mode of operation based on the local ambient conditions was established. Hence the final ANN was trained to decide regarding the mode of operation of the plant and then predict the performance on the basis of ‘switch’ selection for this mode, i.e. ‘0’ for normal mode and ‘1’ for anti-icing mode. However, all the three local ambient conditions are found to be useful for deciding regarding the mode of operation as seen Table 4. The accuracy of prediction for anti-icing was found to be close to 100% when using all three ambient conditions as inputs.

Once the mode of operation was decided, the sensitivity of each of the output parameters was tested for all of these three input parameters and the results of this analysis were summarized in Table 5. It was concluded from Table 5 that though the ambient pressure and temperature have strong effects on the prediction accuracy of selected output parameters, no such dependence of output parameters on the relative humidity was observed. By this sensitivity analysis, relative humidity was considered redundant input for the prediction of performance parameters though it was relevant for deciding the mode of operation. Thus in the final ANN model (refer to Section 4.5) relative humidity was not included as an input parameter for the prediction of the output parameters though this data was required for the selection of mode of operation ‘switch’, i.e. normal or anti-icing mode. Theoretically, the relative humidity does play a role in gas turbine performance

Table 3
Prediction errors by ANN with and w/o anti-icing switch

<table>
<thead>
<tr>
<th>Gas turbine model</th>
<th>Error (%)</th>
<th>$m_{\text{air}}$</th>
<th>$P_{\text{c, out}}$</th>
<th>$T_{\text{c, out}}$</th>
<th>$m_{\text{ng}}$</th>
<th>$T_r$</th>
<th>$P_{\text{el}}$</th>
<th>$\text{CO}_2$</th>
<th>$Q_{\text{ah}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>With anti-icing switch</td>
<td>Average</td>
<td>0.13</td>
<td>0.12</td>
<td>0.09</td>
<td>0.50</td>
<td>0.04</td>
<td>0.21</td>
<td>0.26</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>0.84</td>
<td>0.75</td>
<td>0.52</td>
<td>1.96</td>
<td>0.20</td>
<td>1.23</td>
<td>1.32</td>
<td>5.09</td>
</tr>
<tr>
<td>W/o anti-icing switch</td>
<td>Average</td>
<td>0.50</td>
<td>0.47</td>
<td>0.14</td>
<td>0.79</td>
<td>0.14</td>
<td>0.79</td>
<td>0.81</td>
<td>0.71</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>7.31</td>
<td>6.89</td>
<td>2.21</td>
<td>9.20</td>
<td>2.07</td>
<td>12.18</td>
<td>10.65</td>
<td>6.98</td>
</tr>
</tbody>
</table>

1 Common for all tables showing prediction errors is that all three data sets (i.e. training-, cross validation- and test data set) have been used to calculate the errors. The error levels in the different sets were compared with each other to check for any significant difference between the sets. No such differences were observed, which confirms that the training processes were successful.

Table 4
Sensitivity of prediction of the output for each input

<table>
<thead>
<tr>
<th>Anti-icing model</th>
<th>Correct (no. points)</th>
<th>Misclassified (no. points)</th>
</tr>
</thead>
<tbody>
<tr>
<td>With all inputs</td>
<td>10579</td>
<td>38</td>
</tr>
<tr>
<td>W/o ambient pressure</td>
<td>10492</td>
<td>125</td>
</tr>
<tr>
<td>W/o ambient temperature</td>
<td>10470</td>
<td>147</td>
</tr>
<tr>
<td>W/o relative humidity</td>
<td>10204</td>
<td>413</td>
</tr>
</tbody>
</table>
but since the local variations (during the months of operation) are very limited this input has no effect on the ANN model.

### 4.5. Final ANN model and its prediction accuracy

Based on the results of the sensitivity analysis, the final ANN model for the prediction of gas turbine performance parameters with the input of local ambient conditions was decided as shown in Fig. 3. It was an ANN with two input parameters (i.e. ambient pressure and temperature) for the prediction of performance of the plant connected with a preceding anti-icing ANN to decide about the mode of operation on the basis of three input parameters (i.e. relative humidity in addition to the other two). Thus, though all three input parameters of local ambient were used for the anti-icing ANN the relative humidity data was not required for the performance predicting ANN. However, the mode of operation signal was provided to the later one from the previous one. The accuracy of prediction of power output with this final trained ANN is shown in Fig. 4. The effect of considering anti-icing, for prediction of power output, may be compared with that in Fig. 2 where anti-icing was not considered. Both these predictions are for the same data of power outputs. The errors in predicting all the output parameters by the final trained ANN for a sample of measured data are shown in Table 6. This shows that the errors in prediction of most of the output parameters are very small except two, i.e. gas flow and generated heat. For gas flow prediction, available data with assumed constant calorific value of fuel was used which was not correct. This may be the main reason for the slight inaccuracy in its prediction and accuracy may be improved if measured data for the calorific value would have been available. Regarding the generated heat, this is expected to vary with district heating parameters such as return water temperature. However, in this study the gas turbine was in focus and adding additional input parameters to increase the accuracy of this output parameter was not justifiable. The prediction accuracy is still high enough to validate the inclusion of this output parameter in the final model.

### 5. Development of the graphical user interface

The idea of developing a graphical user interface (GUI) for the ANN model was to make it user-friendly so that any person without the formal knowledge of either the plant or ANN can also use

---

**Table 5**

<table>
<thead>
<tr>
<th>Gas turbine model</th>
<th>Error (%)</th>
<th>$m_{\text{air}}$</th>
<th>$p_{\text{out}}$</th>
<th>$t_{\text{out}}$</th>
<th>$m_{\text{NG}}$</th>
<th>$t_7$</th>
<th>$P_7$</th>
<th>$CO_2$</th>
<th>$Q_{\text{in}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>With all three inputs</td>
<td>Average</td>
<td>0.13</td>
<td>0.12</td>
<td>0.09</td>
<td>0.50</td>
<td>0.04</td>
<td>0.21</td>
<td>0.26</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>0.84</td>
<td>0.75</td>
<td>0.52</td>
<td>1.96</td>
<td>0.20</td>
<td>1.23</td>
<td>1.32</td>
<td>5.09</td>
</tr>
<tr>
<td>W/o ambient pressure</td>
<td>Average</td>
<td>0.73</td>
<td>0.73</td>
<td>0.09</td>
<td>0.91</td>
<td>0.04</td>
<td>0.77</td>
<td>0.80</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>2.97</td>
<td>2.94</td>
<td>0.50</td>
<td>3.71</td>
<td>0.25</td>
<td>3.10</td>
<td>3.14</td>
<td>5.95</td>
</tr>
<tr>
<td>W/o ambient temperature</td>
<td>Average</td>
<td>0.38</td>
<td>0.35</td>
<td>0.12</td>
<td>0.75</td>
<td>0.08</td>
<td>0.59</td>
<td>0.54</td>
<td>0.67</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>2.85</td>
<td>2.80</td>
<td>0.57</td>
<td>4.54</td>
<td>0.69</td>
<td>4.88</td>
<td>4.23</td>
<td>5.08</td>
</tr>
<tr>
<td>W/o relative humidity</td>
<td>Average</td>
<td>0.13</td>
<td>0.11</td>
<td>0.09</td>
<td>0.50</td>
<td>0.04</td>
<td>0.20</td>
<td>0.25</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>Maximum</td>
<td>0.74</td>
<td>0.65</td>
<td>0.45</td>
<td>1.98</td>
<td>0.21</td>
<td>1.12</td>
<td>1.29</td>
<td>4.95</td>
</tr>
</tbody>
</table>
the model to estimate plant performance. The GUI was created in an Excel environment as most people are familiar with this software. As the training of the ANN was over, the weights of this final ANN was used in Visual Basic to generate functions for the prediction of output parameters for a given set of input parameters. The anti-icing ANN was also included in this user interface. The screen image of the user interface is shown in Fig. 5. The entire Excel document is protected with a password except three grey cells (input cells), where the ambient conditions are to be typed in. All the predicted output parameters will be displayed in blue cells (output cells). This display is instantaneous as there is no iteration for this process. Based on input parameters, i.e. ambient conditions, the anti-icing ANN will initially decide regarding the mode of operation which is also displayed in the corresponding blue cell of the GUI. The rest eight output parameters predicted by the trained ANN are also shown in their corresponding blue cells. Two very useful output parameters, power to heat ratio ($a$) and the electrical efficiency ($\eta_e$), of the plant can be calculated using the other predicted parameters and is also calculated in the user interface as shown in Fig. 5. This instantaneous and easy-to-use GUI is found to be very useful for Lund Energii as the operators need not to remember the power or utility, fuel flow or heat outputs for given ambient conditions. This is e.g. helpful in the assessment of gas turbine degradation simply by comparing predictions, which corresponds to a healthy machine, with current measurements. As a next step this GUI can be integrated in the computer system of the plant which enables input signals to be automatically sent to the ANN and generated predictions compared with corresponding measured output signals, demonstrated in [21]. This would realize online condition monitoring of the plant. The GUI can also be used for offline estimation of plant performance for different ambient conditions, helpful e.g. when using weather forecasts in operation planning. Furthermore, since all input parameters are independent of the of the gas turbine, i.e. the gas turbine does not affect the ambient conditions although the ambient conditions affects the gas turbine, opens the possibility of using the model for sensor validation purposes. If, for instance, only one sensor value differs from the corresponding prediction this would indicate a sensor fault rather that an actual fault or degradation.

6. ANN extrapolation capability

Another utility of the developed model was shown for the extrapolation of predictions even beyond the range of training data. The collection of input data for varying ambient conditions can be difficult as it might have to be recorded over the whole year. If the prediction could be done with the trained ANN for those ambient conditions for which data was not available the ANN model could be more useful. This feature has been tested for this developed model. The range of ambient pressure and temperature for which the ANN was trained is shown in Table 7. To examine the extrapolation capacity the predictions for ambient temperatures beyond the range of ANN training data has been compared with actual measured data later. However, the ambient pressure was within the range of the training data set. Thus the extrapolation was tested for ambient temperature, though the same principle is applicable for pressure also. The result for this test is shown in Fig. 6. From Fig. 6 it may be noted that though the training was done for a temperature range of 4.1 °C to –4.5 °C, the extrapolated data prediction was for a range of temperature 5 °C–14 °C.

<table>
<thead>
<tr>
<th>Error (%)</th>
<th>&lt;1</th>
<th>1–2</th>
<th>2–4</th>
<th>&gt;4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass flow rate of air</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Compressor outlet pressure</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Compressor outlet temperature</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mass flow rate of natural gas</td>
<td>6950</td>
<td>548</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Turbine outlet temperature</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Power output</td>
<td>7494</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CO₂ emissions</td>
<td>7452</td>
<td>46</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Generated heat</td>
<td>6328</td>
<td>1051</td>
<td>118</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 5. Screen image of the graphical user interface.

### Table 7

<table>
<thead>
<tr>
<th></th>
<th>Ambient pressure (mbar)</th>
<th>Ambient temperature (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>996.7</td>
<td>4.1</td>
</tr>
<tr>
<td>Minimum</td>
<td>935.7</td>
<td>–4.5</td>
</tr>
</tbody>
</table>
Unfortunately no dataset was available for temperatures below the training interval since temperatures this low are very unusual in this region. The actual measured data, predictions as well as errors in predictions are shown for four output parameters in Fig. 6. This shows that the prediction accuracy is good for ambient temperatures beyond the range of training data. One important conclusion drawn from these results was that the errors in prediction did not increase with increased ambient temperature, more beyond the range of training values of it. The trends of decreasing power output and increasing turbine outlet temperature were predicted well. Thus the extrapolation capacity of the trained ANN for real life systems was found to be possible with this exercise which may be very useful for other similar systems for which the collection of wide range of data for training an ANN may not be feasible due to some practical reasons. In between the period from when the training data and the extrapolation data was gathered an overhaul to some practical reasons. The most important feedback from the plant operators was the usefulness for detection of compressor fouling and thereby optimizing compressor wash intervals. Other possible utilities of this model, e.g. extrapolation of prediction beyond the range of training data, were also demonstrated in this project with good result despite the limitation of an overhauling being performed in between the collection of training data and the data used to check the extrapolation capability.

7. Conclusions

Artificial neural network is found to be a useful tool for prediction of gas turbine performance if it can be trained properly with operational data. This is demonstrated by very high prediction accuracy of the developed ANN model. Developed ANN model may have several utility. It may be used for offline simulation of gas turbine performance or online condition monitoring of the gas turbine for early detection of faults or degradation. It may also be used for sensor validation purposes.

In this paper, the implementation of a demonstration project for the performance prediction of a gas turbine for multi-utility using ANN has been reported. The unified ANN system was trained to first identify anti-icing or normal mode of operation with input of local ambient conditions (pressure, temperature and relative humidity) and then predict the different operating and performance parameters of the gas turbine. To only use one ANN model with ambient conditions as inputs showed to be insufficient as the performance of the gas turbine differs considerably with or without anti-icing in operation. By using an initial ANN, to decide regarding the switching on or off of the anti-icing system, and providing information to the second ANN about the mode of operation the prediction accuracy was increased considerably.

A user-friendly interface was also developed for this ANN to display the predictions instantaneously along with calculated values of power to heat ratio and electrical efficiency of the plant. This can be used for online monitoring as well as offline estimation of expected performance of the plant with varying local ambient conditions. The GUI may be more useful for industrial use as it does not need any special skill to use it once the training is over.

The most important feedback from the plant operators was the usefulness for detection of compressor fouling and thereby optimizing compressor wash intervals. Other possible utilities of this model, e.g. extrapolation of prediction beyond the range of training data, were also demonstrated in this project with good result despite the limitation of an overhauling being performed in between the collection of training data and the data used to check the extrapolation capability.

References

[21] Fast M, Assadi M, Smrekar J. Application of artificial neural network to the condition monitoring and diagnosis of a CHP plant, ECDS; accepted for publication.
Application of artificial neural network to the condition monitoring and
diagnosis of a combined heat and power plant
Fast M, Palmé T.
Application of artificial neural networks to the condition monitoring and diagnosis of a combined heat and power plant

M. Fast a,*, T. Palmé b

a Division of Thermal Power Engineering, Department of Energy Sciences, Lund University, P.O. Box 118, S-221 00 Lund, Sweden
b Department of Mechanical and Structural Engineering and Materials Science, University of Stavanger, N-4036 Stavanger, Norway

Abstract

The objective of this study has been to create an online system for condition monitoring and diagnosis of a combined heat and power plant in Sweden. The system in question consisted of artificial neural network models, representing each main component of the combined heat and power plant, connected to a graphical user interface. The artificial neural network models were integrated on a power generation information manager server in the computer system of the combined heat and power plant, and the graphical user interface was made available on workstations connected to this server.

The plant comprised a Siemens SGT800 gas turbine with a heat recovery steam generator as well as a bio-fueled boiler and its steam cycle. Steam from the heat recovery steam generator and the bio-fueled boiler expanded together in a common steam turbine, producing both electricity and heat. The artificial neural network models were trained with operational data from the components of the combined heat and power plant.

Accurate predictions from the ANN (Artificial neural network) models in combination with an undemanding integration in the power plant’s computer system were some of the main conclusions from this study.

1. Introduction

The main interest of the present research includes an increased availability, a reduced maintenance cost as well as a more efficient maintenance, i.e., condition-based maintenance. By linking historical operational data to the analysis of plant condition, an optimization of maintenance and performance of a plant becomes possible. Deviations from an expected data pattern could indicate for instance a faulty component or degradation, whereupon warnings and/or alarms can be generated.

The plant studied is a hybrid combined heat and power (CHP) plant consisting of a gas turbine with its heat recovery steam generator and a biomass fueled boiler with its steam cycle. Artificial neural network (ANN) is the data modeling tool used for this study and operational data the basis for training the networks.

Several authors have reported on a variety of applications, such as monitoring, diagnosis, optimization, fault diagnosis etc., by means of data driven modeling. In [1] ANN is employed for the optimization of load allocations and in [2] a fault diagnosis neural network is trained with operational data from a nuclear power plant. Other examples include [3,4] where ANN is used for diagnosis of a turbofan engine and fault diagnosis in a power plant. Researchers at the Division of Thermal Power Engineering at Lund University have carried out several investigations – References [5–8] representing some of them – in this field and ANN has proven to be a good candidate for the modeling of non-linear energy systems.

The CHP plant components were simulated with ANN and the models were integrated on a power generation information manager (PGIM) server in the computer system of the CHP plant. The PGIM server continuously receives signals from the plant and also stores the signals for up to seven years. The ANN models are fed with measured signals in real time to provide predictions of the expected plant performance, and these predictions are presented through a graphical user interface (GUI) and are also stored on the server in the event that the plant’s performance history needs to be evaluated. Each time new predictions are calculated, a comparison between these and the corresponding measured signals is made, whereupon warnings and/or alarms are generated for possible anomalies. The warning and alarm limits can be individually set for all output parameters, according to, for example, operator demands or ANN and sensor accuracy. If several warnings and/or alarms occur it could be an indication of degradation or component malfunction. In the case of a single warning or alarm, a faulty sensor may be the cause. Furthermore, production costs and income are
continuously calculated on both measurements and predictions, using current electricity prices, taxes, fuel prices etc., and enables an economic evaluation of plant operation and maintenance.

The integration of the ANN models and the accompanying GUI, in the computer system was carried out successfully. With the developed tool, the plant condition could be monitored in real time while possible deviations, such as degradation, were simultaneously put through an economical evaluation. The prediction-based economic calculations demonstrated very good results, much on account of the very high prediction accuracy of the ANN models.

2. The CHP plant “Västhamnsverket”

The plant, Västhamnsverket, is located in Helsingborg, Sweden, and is owned by Öresundskraft AB. It is a unique multi-fuel CHP plant comprising a SGT800 gas turbine, a heat recovery steam generator (HRSG) and a steam cycle with a biomass fired boiler, a steam turbine, pre-heaters and two district heating condensers. The boiler was installed by Götaäverken Ångteknik, Sweden, in 1982. In the year 1999, a gas power plant, made up of a gas turbine and a HRSG, was added to the existing steam plant. As a result, the CHP plant presently operates as a ‘hybrid’ plant. It generates steam from the waste heat of the gas turbine in the HRSG as well as in the boiler using biomass as fuel. The steam from the HRSG and the boiler expands in one common steam turbine. The basic schematics of the plant are shown in Fig. 1. By adding a gas power cycle to the existing steam cycle, the total power output and district heat could be significantly increased. The system solution was unique and demonstrated a high alpha-value (power to heat ratio) of 0.68. In total, approximately 125 MWe and 186 MWth are produced [9,10].

The SGT800 gas turbine (formerly known as GTX100) was manufactured by Siemens Industrial Turbomachinery AB in Sweden. It was designed to produce up to 44 MW of power with a thermal efficiency of 37%, and is optimized for combined cycle operation, which leads to a relatively high exhaust gas temperature being maintained. This particular gas turbine runs at full load during most of its operation time, otherwise its power output is set to 30 MW [10].

3. Basics of ANN and its training

ANN is a non-linear statistical data modeling tool that basically learns from experience; it is, in other words, adaptive [11]. Instead of being built a priori from specification, neural and adaptive systems use external data to automatically set their parameters [12]. ANN can be used to solve a variety of tasks, including classification, regression, general estimation problems, etc. An ANN consists of a group of interconnected artificial neurons processing information in parallel. The performance of a network can be improved by rendering it “aware” of its output(s) through a performance feedback loop that includes a cost function. The feedback is used to adjust the network parameters through systematic procedures called learning or training rules, in order to improve the system output with respect to the desired goal [12].

The most common feedback algorithm is the back-propagation algorithm, proposed by Rumelhart et al. in 1986 [13], which provides a computationally efficient method for evaluating the error function associated to each weight. The updating algorithm is based on a gradient descent method, and the weights are updated in order for the error function to be minimized. The multi-layer feed-forward network, more specifically the multi-layer perceptron (MLP), was the network type employed during this study. It consists

### Nomenclature

<table>
<thead>
<tr>
<th>C</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Income</td>
</tr>
<tr>
<td>aux</td>
<td>Auxiliary</td>
</tr>
<tr>
<td>d</td>
<td>Delivery</td>
</tr>
<tr>
<td>e</td>
<td>Electricity</td>
</tr>
<tr>
<td>EP</td>
<td>Emission permits</td>
</tr>
<tr>
<td>GC</td>
<td>Green certificates</td>
</tr>
<tr>
<td>m</td>
<td>Maintenance</td>
</tr>
<tr>
<td>NG</td>
<td>Natural gas</td>
</tr>
<tr>
<td>prod</td>
<td>Production</td>
</tr>
<tr>
<td>r</td>
<td>Return</td>
</tr>
<tr>
<td>th</td>
<td>Thermal</td>
</tr>
</tbody>
</table>

### Abbreviations

- ANN: Artificial neural network
- CHP: Combined heat and power
- DH: District heating
- FW: Feedwater
- GUI: Graphical user interface
- HP: Heat pump
- HRSG: Heat recovery steam generator
- HRU: Heat recovery unit
- MLP: Multi-layer perceptron
- MSE: Mean squared error
- PGIM: Process generation information manager
- PH: Pre-heater
- SEK: Swedish Krona
- SH: Super heater
- VIGV: Variable inlet guide vanes

---

Fig. 1. Västhamnsverket configuration.
of an input layer, one or more hidden layer(s) and an output layer. However, there is no impediment to having more than one hidden layer, since it has been proven that one layer with hidden neurons is enough to approximate any continuous function if it only has a sufficient number of neurons [14]. Once the inputs are presented to the network, they will be multiplied by their adjustable weights and in each processing element summed and passed through a transfer function in order to produce outputs. The data used as inputs are transmitted through the network, layer by layer, and a set of outputs is obtained. Errors are generated by comparing these outputs with those desired and they are then used for updating the weights before another set of inputs is transmitted through the network. This is referred to as supervised training. The mean squared error (MSE) is calculated between each epoch (iteration) and the training is terminated when the MSE is satisfactorily low. At this point, the weights are fixed and the network is ready to predict outputs from previously unseen data, known as generalization.

Prior to the commencement of the training process, the data set is randomized and divided into three subsets: training, cross-validation and test sets. The training set is used by the network for weight adjustment (i.e., learning) whereas the cross-validation set is used to avoid overtraining by the network. Finally, the test set is used after the training to ensure that a decent generalization capability is obtained. A properly trained network displays similar prediction errors, or accuracy, in all three data sets.

Even though ANN is adaptive, the actual network configuration has to be specified. A selection of the number of hidden neurons is carried out by a trial-and-error procedure where the network is trained multiple times with a varying number of neurons in the hidden layer. The network exhibiting the highest prediction accuracy, in all three data sets, is selected. The number of neurons required in the hidden layer is determined by the complexity of the input/output parameter dimension. It is preferable to select as few neurons as possible, without sacrificing the prediction accuracy, in order to obtain good generalization capabilities. Each neuron includes a transfer function, in which the incoming data is processed. In this study, based on accumulated experience at the gas turbine, HRSG, boiler and steam turbine, and each component was modeled separately. Data from the hybrid plant was delivered as 5-min averages, covering three months of operation. A baseline was established and all data recorded after this was considered as healthy and thereby suitable for ANN training. However, before using any data for training it had to be filtered and outliers, etc., removed. Also all transient operations were removed since 5-min average data only permitted modeling of the steady state operation. The selection of input and output parameters, for each individual model, was based on the availability of reliable plant data as well as true needs. All ANN models were subjected to a sensitivity analysis in order to assess which input parameters were of significance for each model.

Sections 4.1–4.4 address the structure and prediction performance of the ANN models. The prediction performance was tested with a number of unseen data patterns and the prediction errors were divided into four groups, i.e., errors below 1%, 1–2% errors, 2–4% errors and errors above 4%.

### 4. ANN modeling of the CHP plant

The system was divided into its basic components, i.e., the gas turbine, HRSG, boiler and steam turbine, and each component was modeled separately. Data from the hybrid plant was delivered as 5-min averages, covering three months of operation. A baseline was established and all data recorded after this was considered as healthy and thereby suitable for ANN training. However, before using any data for training it had to be filtered and outliers, etc., removed. Also all transient operations were removed since 5-min average data only permitted modeling of the steady state operation. The selection of input and output parameters, for each individual model, was based on the availability of reliable plant data as well as true needs. All ANN models were subjected to a sensitivity analysis in order to assess which input parameters were of significance for each model.

Sections 4.1–4.4 address the structure and prediction performance of the ANN models. The prediction performance was tested with a number of unseen data patterns and the prediction errors were divided into four groups, i.e., errors below 1%, 1–2% errors, 2–4% errors and errors above 4%.

#### 4.1. Gas turbine model

When operating at full load, the performance of a gas turbine is determined by the ambient conditions. Hence, using these conditions as input parameters to the ANN model is a natural course of action. However, since the gas turbine was set to run at either full load or to a limit of 30 MW, an input parameter representing these modes was necessary. The two discrete load cases were represented by two ‘switches’, i.e., ‘1’ and ‘0’, enabling the neural network to differentiate between the two modes of operation based on load. Another real-life issue was that of harmful ice
formation in the intake manifold during local ambient conditions that were simultaneously cold and humid. To avoid this, an anti-icing system was utilized to preheat the air before the compressor. The anti-icing operation also constituted a discrete mode and was represented by another set of ‘1’ and ‘0’ switches in the ANN model.

A complete list of input and output parameters for this ANN is presented in Table 1, and the ANN prediction accuracy is given in Table 2. Fig. 2 shows a comparison between measured and predicted power together with the prediction error. Generally very good prediction accuracy is attained with the gas turbine model and most errors are below 1%. The VIGV (variable inlet guide vane) angle and the fuel flow are exhibiting somewhat elevated error levels in their prediction. The reason for the slightly higher errors in gas flow predictions can be referred to the fact that no consideration to variations in the calorific value of the fuel is taken. Also, during the collection of the data the control of the VIGVs was experiencing small fluctuations which are the reason behind the higher prediction errors for this parameter.

4.2. HRSG model

The input parameters for the HRSG ANN were very similar to the ones for the gas turbine ANN due to the fact that the gas turbine governed the performance of the HRSG. The difference was constituted by the fact that the HRSG was also affected by the district heating circuit and the feedwater properties. A complete list of input and output parameters for the HRSG model can be seen in Table 3 and the prediction accuracy of the model is provided in Table 4. The temperature predictions are very good whereas flow and heat predictions have a lower accuracy. This difference is most likely the cause of flows being generally harder to measure with good accuracy and the fact that sensors located on the district heating circuit are not prioritized as high and thereby not calibrated regularly. All predictions are, however, sufficiently accurate.

4.3. Boiler model

Only two input parameters, i.e., the temperature and pressure of the feedwater, were required for the ANN boiler model in order to obtain predictions of the steam properties and the mass flow rate of pellets. The in- and output parameters of the boiler model are listed in Table 5 and the error distributions for the predictions are presented in Table 6.

4.4. Steam turbine model

The prediction of the power output and the produced heat in the district heating condensers requires a number of input parameters.

The most obvious of these are the steam properties, both from the HRSG and the boiler, along with return and delivery temperatures of the district heating. The heat generated in the boiler economizer and the heat pump are added as inputs since they are located on the same district heating circuit as the condensers and thereby also affect the system. Table 7 shows the model structure while Table 8 and Fig. 3 illustrate the prediction performance of the model. The power output and the feedwater temperature are predicted very well while the condenser heat exhibits larger prediction errors. The condenser heat is on the other hand calculated from several other measured parameters and this behavior is, therefore, expected.

5. Economic calculations

The idea behind the economics part is to always have two values of the production cost, i.e., actual and “predicted” values, thus allowing for instance an economical evaluation of the plant degradation. This is possible since the ANN models always predict the performance of a healthy plant and economic calculations based on these predictions will indicate what the production cost should be. Economic factors that are taken into account include fuel prices and taxes on fuels and energy production, etc. The production cost is divided between the two production types, i.e., gas turbine/natural gas and solid fuel boiler/pellets. Since these are closely connected through the hybrid configuration, special consideration is needed when calculating the production cost.

Sections 5.1–5.3 briefly describe three types of control mechanisms applied for the energy sector in Sweden. Furthermore, Sections 5.4 and 5.5 explain the basics for calculating the production cost for the two production types.

5.1. Carbon dioxide tax

In order to reduce CO2 emissions, all fossil fuels, such as coal, oil and natural gas, are subject to a CO2 tax. However, electricity production is exempt of the tax and heat produced in combined heat and power plants is only subject to 21% of it. Pure heat production, on the other hand, is subject to full, 100%, CO2 tax. Various fuels generate different amount of CO2 for which reason the CO2 tax varies depending on the fuel type [15].

5.2. Emission trading

The emission trading is a means of reducing the emission of greenhouse gases into the atmosphere, in accordance with the
Kyoto protocol. At the present time, only CO₂ is included in the system, but more greenhouse gases could be added in the future. The government sets a limit or cap on the accepted emissions of a certain pollutant. Companies or other groups are issued emission permits and are required to hold an equivalent number of allowances (or credits) that represent the right to emit a specific amount. The total amount of allowances and credits cannot exceed the cap, thus limiting total emissions to that level. Companies that need to increase their emissions must buy credits from those who pollute less. In effect, the buyer thus pays a charge for polluting, while the seller is being rewarded for having reduced emissions by more than what is required [15,16].

5.3. Green certificates

Producers of electricity are awarded one green certificate for every megawatt hour of electricity generated with renewable energy sources. "Renewables" include wind, solar, wave and geothermal energy along with certain types of bio-fuels and hydro energy. The demand for green certificates is created through a quotation system that states the proportion of renewable sources in the energy mix from year to year [15].

5.4. Gas turbine/natural gas

Four larger posts are used to calculate the production cost, i.e., fuel costs, tax costs, cost of emission permits and income of electricity. The smaller posts are maintenance costs and tax costs for auxiliary power used for electricity production. Everything is calculated in Swedish Krona (SEK) per megawatt hour of produced heat according to Eq. (1).

\[
C_{\text{prod}} \left[ \frac{\text{SEK}}{\text{MWh}_{\text{th}}} \right] = C_{\text{NG}} + C_{\text{tax}} + C_{\text{EP}} + C_{\text{m}} + C_{\text{aux}} - I_e
\] (1)

When calculating the production cost, all plant specific parameters are available both as measurements and predictions, thereby enabling a comparison of an "optimal" production cost with the actual value. Examples of parameters used in the calculations are fuel flow, condenser heat and power output.

5.5. Solid fuel boiler/pellets

Three larger posts and one smaller post are used to calculate the production cost. The larger posts include fuel costs, income of electricity and income of green certificates. The smaller post concerns the maintenance costs.
6. The graphical user interface

The ANN models are integrated on a PGIM server and continuously receive the latest operational data in order to generate predictions. These are then stored on the PGIM server. Both operational data as well as historical and current predictions are accessible through the GUI, which in return is accessible on all workstations connected to the PGIM server. In order to be foreseeable and easy to use, the GUI is developed in Excel. It is divided into five sheets of which the first displays an overview of the entire hybrid plant showing the main parameters for each component together with the production cost for the gas turbine and the solid fuel boiler. Various inputs to the system, both thermodynamic and economic, are also shown. The ANN predictions are displayed below their equivalent measured values, which provide an evaluation of the plant performance. The GUI automatically warns or alerts the operator for possible deviations from the expected data pattern. The GUI update frequency is adjustable to fit user needs and at each update, the latest values are communicated from the PGIM server. The remaining four Excel sheets contain detailed information concerning the specific components of the hybrid plant, i.e., the gas turbine, HRSG, solid fuel boiler and steam turbine. Economical inputs are given by the user, with the exception of the price of electricity which is updated automatically with data from Nord Pool [Nordic power exchange] [17]. Fig. 4 shows the GUI plant overview and Fig. 5 shows the GUI gas turbine view.

To alert the operator of deviations between measurements and predictions, a warning/alarm indicator is located on the main view of the GUI. Limits for warnings and alarms are individually specified for each parameter depending on, for example, measurement and prediction accuracy.

With the developed tool for parameter analysis, the user is, with a few clicks, able to analyze any chosen parameter for any chosen time interval. This is possible since the predictions are stored together with measurements on the PGIM server. After selection of a parameter and interval, a plot with measurements and predictions is automatically generated. This is demonstrated in Fig. 6.
7. Conclusions

The present study has demonstrated that ANN modeling of all the major CHP plant components is possible. With proper training, data and parameter selection, it is also feasible to achieve very high prediction accuracies. It was shown that a real product, for online condition monitoring, can be developed and integrated on-site, using an artificial neural network. The graphical user interface was accessible through all workstations connected to the internal network, providing a capacity for online monitoring over the entire site. With the developed tool, the condition of a plant could thus be monitored while simultaneously economically evaluating possible deviations, such as degradation or faults. Hence, by using the developed tool, an optimization of the plant operation and maintenance was rendered possible.

The described ANN models are plant specific; however, the method is general and thereby applicable to other power plants and power plant configurations.

Acknowledgements

The funders of this project, the Swedish Gas Center AB (SGC) and Värmeforsk AB, are greatly acknowledged. Siemens Industrial Turbomachinery AB and Öresundskraft AB are also greatly acknowledged for their assistance regarding gas turbine, power plant and computer system related matters.

References


[9] Hellberg C. Distribution of electricity and heat production between the powder boiler and the heat recovery steam generator. [Fördelning av el- och värmeproduktion mellan pulverpannan och avgaspannan]. Master’s thesis. Lund: Division of Thermal Power Engineering, Department of Energy Sciences, Lund University (P.O. Box 118, S-221 00); 2004 [in Swedish].


Condition based maintenance of gas turbines using simulation data and artificial neural network: A demonstration of feasibility
Fast M, Assadi M, De S.
Presented at ASME Turbo Expo 2008, Berlin, Germany
ABSTRACT

Gas turbine maintenance is crucial due to high cost for the replacement of its components and associated loss of power during shutdown period. Conventional scheduled maintenance, based on equivalent operating hours, is not the best alternative as it can require unnecessary shut downs. Condition based maintenance is an attractive alternative as it decreases unnecessary shut downs and has other advantages for both the manufacturers and the plant owners. However, this has shown to be a complex/difficult task. A number of methods and approaches have been presented to develop condition monitoring tools during the past decade. Condition monitoring tools can e.g. be developed by means of training artificial neural networks (ANN) with historical operational data. Such tools can be used for online gas turbine performance prediction where input data from the plant is fed directly to the trained ANN models. The predicted outputs from the models are compared with corresponding measurements and possible deviations are evaluated. With this method both recoverable degradation, caused by fouling, and irrecoverable degradation, caused by wear, can be detected and hence both compressor wash and overhaul periods optimized. However, non-availability of operational data at the beginning of the gas turbine operation may cause problems for the development of ANN based condition monitoring tools.

Simulation data, on the other hand, may be generated by using a manufacturer’s engine design program. This data can be used for training artificial neural networks to overcome the problem of non-availability of operational data. ANN models trained with simulation data could be used to monitor the engine from the very beginning of its operation. A demonstration case using a Siemens gas turbine has been shown for this proposed method by comparing two ANN models, one trained with operational data and the other with simulation data. For the comparison an arbitrary section of operational data was used to produce predictions from both models, whereupon these were plotted with corresponding measured data. The comparison shows that the trends are very similar but the parameter values for the measured and the simulated data are shifted by a constant. Using this knowledge, one can provide an ANN based engine monitoring tool that could be adjusted to a certain engine using engine performance test data. The study shows promising results and motivates further investigations in this field.

1 INTRODUCTION

Gas turbines for power generation need continuous monitoring and proper maintenance as their sudden break downs may lead to heavy penalty both due to expensive spare parts and the production loss accumulated. This is even more stringent now under the changed scenario of deregulation of electricity. Conventionally original equipment manufacturers (OEM) recommend scheduled maintenance of gas turbines. Intervals of scheduled maintenance are based on the equivalent operating hours (EOH) [1]. Warranty of gas turbines is usually subjected to compliance of scheduled maintenance by
the plant owner. Thus the plant owner is forced to shut down its operation for inspection and routine maintenance, loosing power production during these periods. In addition to this the inspections themselves are also very costly. However, the OEMs are increasingly offering a “service-based” approach to marketing their products, in which their customers are guaranteed certain availability of the engine after purchase [2]. This means that the OEMs are now sincerely interested in condition based maintenance since they directly benefit from delaying any potential service as long as possible. This “service-based” approach is also a method of keeping third party manufacturers, selling cheaper spare parts, out of the market.

Modern gas turbines are equipped with sensors measuring many operational parameters. Some of the measured parameters are used directly by the gas turbine control system. The measured values are also stored in a database, useful when e.g. unexpected gas turbine operation needs to be examined. OEMs use equation based programs to simulate the performance of their gas turbines and to draw conclusions about its health or for other purposes. These programs are based on heat- and mass balances, compressor maps etc., and require iterations to converge to a solution. Such programs also require expert knowledge and man hours to run them and make useful conclusions.

To achieve condition based maintenance the OEMs need a condition monitoring tool that is fast, simple, reliable and cheap for online monitoring of the gas turbine. Any unexpected change in gas turbine behavior needs to be reported back to the OEM instantly. Acceptable limits for parameter deviation can be set individually for each parameter and imbedded in the tool.

Artificial neural network (ANN) can be used to develop such continuous condition monitoring tool, using already stored data, without any additional significant expenditure [3][4]. The ANN predicts values of desired parameters instantly without any iterations and hence, suitable for online applications. ANN may be trained with data captured during the healthy baseline operation of the turbine. This trained ANN predicts the expected output from a healthy turbine when provided with input parameters from the plant. Any reasonable difference observed online between the predictions by this trained ANN and the actual measured values of parameters in the plant are indications of a fault or degradation of the equipment, either recoverable or irrecoverable. The maintenance operation would be decided accordingly. However, non-availability of operational data for the training of the ANN is a problem at the beginning of the operation. Accumulation of sufficient data for different operational condition of the gas turbine is also a very time consuming process. In order to solve this problem, the use of simulation data from the OEM’s design program is proposed. These programs could be used to generate data in the resolution and intervals needed for ANN training. It is expected that the data from the design program of the OEM would grossly match with that of the initial gas turbine operation as it has not yet degraded. Using simulation data enables integration of an ANN based condition monitoring tool in the control system of new gas turbines. This could benefit the plant owner through increased availability and the OEM through saving in maintenance.

In this work, feasibility of this proposition of condition based maintenance of gas turbines has been examined and results are discussed. Two ANNs were trained to predict the performance of a gas turbine. One was trained with simulation data from the OEM’s design program and the other was trained with operational data from the plant. Comparison of predictions by these two ANNs and subsequent analysis proved that an ANN trained with simulation data could solve the problem of non-availability of operational data. However, these two ANNs were not developed for this specific purpose. Their predictions were only used to check the feasibility of condition based maintenance from the beginning of gas turbine operation. As a result, comparison of predictions by these two ANNs was carried out for equivalent conditions of operation with proper reasoning. This showed that the ANN trained with simulation data could be used for condition monitoring and thus the feasibility of condition based maintenance of gas turbines is well established.

2 THE PLANT AND GAS TURBINE

The plant, Västhamnsverket, is located in Helsingborg, Sweden and is owned by Oresundskraft AB [5]. It is a unique multi-fuel combined heat- and power (CHP) plant. The plant consists of a gas turbine, a heat recovery steam generator (HRSG) and a steam cycle consisting of a biomass fired boiler, a steam turbine, pre-heaters and two condensers. These are the main components of the plant as shown in Figure 1.

![FIGURE 1. SCHEMATIC LAYOUT OF THE CHP PLANT [6]](image)

The boiler was installed by Götaverken Ångteknik, Sweden in 1982. In the year 1999, a gas power plant consisting of a gas turbine and a HRSG was added to the existing steam plant. Thus the CHP plant presently operates as a ‘hybrid’ multi-fuelled plant. It generates steam from the waste heat of the gas turbine in the HRSG as well as in the boiler using biomass as fuel. The steam from the HRSG and the boiler expands in one common steam turbine. By adding a gas power cycle to the existing steam cycle, total power output and district heat were increased significantly. The system solution is unique and has a high alpha-value (power to heat ratio) of 0.68. In total approximately 125 MW_e and 186 MW_th are produced. [7][8]

The SGT800 (formerly known as GTX100) gas turbine was manufactured by Alstom Power AB (presently Sie-
The gas turbine was designed to produce up to 44 MW of power with a thermal efficiency of 37%. It is a single-shaft engine with the generator on the ‘cold’ side (i.e. cold end drive). The compressor consists of 15 stages with a pressure ratio of around 20 and a mass flow of 130 kg/s. In the combustion chamber 120 MW fuel is burned, generating 100 MW of turbine work. Due to the very high temperatures after the combustion chamber the first two stages of the turbine are cooled with compressor air. This gas turbine is optimized for combined cycle operation and therefore a relatively high exhaust gas temperature is maintained. [8]

The SGT800 gas turbine at Västhamnsväxket was the production prototype, co-owned by Öresundskraft AB and Siemens Industrial Turbomachinery AB, and has been used for component test and development. Due to this it has been subject to many component exchanges etc., why a change in behavior during its years of operation might be expected.

The gas turbine runs at full load during most of the time of its operation otherwise its power output is set to 30 MW. The gas turbine is limited to a lower power output when only the generated heat is desired. It is however only economically beneficial to operate the gas turbine during the winter months due to local conditions, such as ambient temperature, electricity and gas prices. The gas turbine is also equipped with an anti-icing system which preheats the inlet air to avoid harmful ice formation during simultaneous cold and humid conditions. The anti-icing system is normally in operation when the ambient temperature is somewhere between +5 °C and -5 °C while at the same time the relative humidity exceeds 80%. When preheating the air the gas turbine efficiency and power output decreases due to decreasing air density. In this power plant the inlet air for the gas turbine is preheated in a heat exchanger using district heating water.

Till May 2007, fifty two such SGT800 machines were sold, out of which twenty four were in commercial operation. Two of these machines have more than 45 000 EOH and seventeen machines have more than 20 000 EOH. Out of the total fifty two machines, ten machines are operating in gas turbine power plants, seventeen in co-generation plants and twenty five in gas-steam combined power plants.

3 ANN MODEL BASED ON SIMULATION DATA

This first SGT800 ANN model was developed in 2001. The data was generated with Siemens gas turbine design program using an allocated performance deck license. The design program is based on traditional heat- and mass balances but also contains the characteristics describing the SGT800 gas turbine. The ANN model was developed to show the preliminary feasibility of ANN for simulation of gas turbine performance. The input and output parameters of the ANN were decided accordingly. There were only a few input and output parameters as shown in Table 1. However, the accuracy of predictions by this ANN was acceptable. [9][10]

The ANN was constructed with the multi-layer feed-forward network type and trained using back-propagation. The data was divided into three sets, one set for training, another set for cross-validation (to check for overtraining) and a final set for independent testing after the training was completed. Furthermore, one hidden layer was chosen and the network was optimized regarding the number of neurons in this layer. Three training runs were performed for each network configuration with a maximum of 10 000 epochs for each run. For this work Matlab neural toolbox was used. [9][10]

<table>
<thead>
<tr>
<th>TABLE 1. INPUT AND OUTPUT PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inputs</strong></td>
</tr>
<tr>
<td>Load [%]</td>
</tr>
<tr>
<td>Ambient temperature</td>
</tr>
<tr>
<td>Relative humidity</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

4 ANN MODEL BASED ON OPERATIONAL DATA

The second SGT800 ANN model was developed using operational data from the plant. Training data for this ANN was collected during one month of operation in 2006. This period was just after an overhauling of the turbine. During this overhauling components were inspected and the compressor was washed. This corresponds to the baseline and data used for training was considered to represent healthy and clean condition of the turbine. Only the steady state operation of the turbine was modeled since the resolution of data did not allow any transient operation to be modeled. The output parameters of this model were different from those of the first one since they were decided based on ‘real life’ needs and the availability of reliable plant data for modeling. As mentioned earlier, this particular turbine operates either at full load or at 30 MW. Hence, instead of using load as an input, the two discrete load cases were represented by two ‘switches’, i.e. ‘1’ and ‘0’. This enabled the neural network to differentiate between the two modes of operation based on load. Another ‘real life’ issue was that of harmful ice formation in the intake manifold during simultaneous cold and humid local ambient conditions. To avoid this, an anti-icing system is used to preheat the air before the compressor. This anti-icing operation is also a discrete mode and represented by another set of switches of ‘1’ and ‘0’ in the ANN model. A complete list of input and output parameters for this ANN is shown in Table 2. The use of relative humidity as an input was redundant since its effects on gas turbine performance was marginal. It should be reminded however, that the relative humidity has large effect on ice formation. This ANN model also proved to be successful in prediction of all output parameters with very small errors. [11]
This neural network was trained using the same approach as the neural network based on simulation data. However, for this work the commercial software NeuroSolutions was used. [11]

### TABLE 2. INPUT AND OUTPUT PARAMETERS

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation mode</td>
<td>Power output</td>
</tr>
<tr>
<td>Anti-icing mode</td>
<td>Compressor inlet pressure</td>
</tr>
<tr>
<td>Ambient pressure</td>
<td>Inlet guide vanes angle</td>
</tr>
<tr>
<td>Ambient temperature</td>
<td>Bleed temperature</td>
</tr>
<tr>
<td></td>
<td>Compressor outlet pressure</td>
</tr>
<tr>
<td></td>
<td>Compressor outlet temperature</td>
</tr>
<tr>
<td></td>
<td>Mass flow rate of fuel</td>
</tr>
<tr>
<td></td>
<td>Mass flow rate of air</td>
</tr>
<tr>
<td></td>
<td>Exhaust gas temperature</td>
</tr>
</tbody>
</table>

### 5 COMPARISON OF PREDICTIONS

The objective of this study was to examine the feasibility of using simulation data from the OEM’s design program for the ANN model development to solve the problem of non-availability of operational data at the beginning of operation. The two developed ANN models represent the same gas turbine, one based on simulation data and the other based on operational data. Therefore it was of interest to compare the predictions from these two ANN models in order to assess the usefulness of the model based on simulation data. This was done by selecting an arbitrary section of operational data. Values of input parameters from the plant data was presented to both models and values of common outputs parameters were compared. Data used for the comparison was previously unseen to the models and represents roughly eighteen days of operation at full load. Predictions of power output and exhaust gas temperature by these two neural networks along with measured values are shown in Figures 3 and 4 respectively. The first ANN model, based on simulation data is referred to as ANN 1 and the second model, based on operational data, as ANN 2. On the X-axis the number of data points was used as parameter where each point represents a five minute average value. As expected, the ANN model based on operational data predicted the power output and the exhaust gas temperature with better accuracy. The predictions from the ANN model based on simulation data were not as impressive apparently. The difference in predictions by these models was however expected since the gas turbine had been operating for several years while the simulation data represented the performance of a new gas turbine. The ANN based on simulation data predicted higher power outputs than those actually measured and for exhaust gas temperature it was just the opposite, as shown in Figures 3 and 4. During the years, 2001 till 2006, when simulation data was generated (2001) and the operational data was collected (2006) several overhauls, replacements of components etc. occurred. The gas turbine performance degradation was thus expected which partly corresponds to the results shown in these figures. To compare the trends of predictions by these two ANNs, plots of predictions by one model is shifted vertically to check the matching of these two plots. These plots are shown in Figures 5 and 6. Obviously for these figures, plots of predictions by the two ANNs are represented by different axes in vertical directions as shown. Even for these plots, predictions by the two ANNs did not overlap. Apparently this seemed to be a failure of the use of simulation data. However, it was revealed that the mode of operation changed with respect to anti-icing during this period. There was no consideration of anti-icing in the simulated data. Moreover, effect of ambient pressure was not considered in the simulated data either. Thus comparison of predictions by the two models during normal mode of operation without anti-icing would be more meaningful. It was identified that a section of data points, i.e. data points 4050 to 5200, were during operation without anti-icing. Plots were made for these data points only as shown in Figures 7 and 8 with vertical shifting similar to Figures 5 and 6. In these plots, predictions by the two models match very closely for exhaust gas temperature. However, there still exists some difference for power output predictions. The reason behind this difference is due to the consideration of ambient pressure as an input parameter for the second model though it was not used for the first one. Variations of ambient pressure do not affect the exhaust gas temperature significantly though it was expected to influence the power output. This can be demonstrated through adjusting the power output predictions from ANN 1 according to variations in ambient pressure by applying Equation 1. The results are shown in the final figure, Figure 9, and confirm that the difference in power output predictions (seen in figure 7) between the two ANNs is due to the fact that ambient pressure was not included as an input to the first ANN (based on simulation data). Index i in Equation 1 represents the data points, $p_{ISO}$ is the constant atmospheric ISO pressure (1,013 bar).

$$P_{adj, i} = P_{ANN, i} \cdot \frac{p_{ambient, i}}{p_{ISO}}$$  

Thus it was concluded that the model developed using simulation data was equivalent to that using operational data under identical operational condition. However, the vertical shift in prediction by one model was partly due to the degradation of the plant as the second model was developed using data after about five years of operation. Another reason for the vertical shift is the fact that this gas turbine was, as mentioned earlier, the production prototype and thereby subject to many component exchanges etc. However, interesting to notice is, despite the circumstances, that the generalization capability of the ANN based on simulation data is very accurate. This indicates that no retraining of ANN models is needed due to e.g. overhauls or component exchanges. This study was successful to establish the possibility of using simulation data for developing ANN models for condition based maintenance of gas turbines from the beginning of its operation. Furthermore, the results of this study add value to previous studies regarding fault diagnosis tools based on ANN and simulated faults [10].
FIGURE 3. PREDICTIONS OF POWER OUTPUT

FIGURE 4. PREDICTIONS OF EXHAUST GAS TEMPERATURE
FIGURE 5. PREDICTIONS OF POWER OUTPUT WITH VERTICAL SHIFT

FIGURE 6. PREDICTIONS OF EXHAUST GAS TEMPERATURE WITH VERTICAL SHIFT
FIGURE 7. PREDICTIONS OF POWER OUTPUT WITH VERTICAL SHIFT AND W/O ANTI-ICING

FIGURE 8. PREDICTIONS OF EXHAUST GAS TEMPERATURE WITH VERTICAL SHIFT AND W/O ANTI-ICING
6 CONCLUSIONS

Introduction of condition based maintenance instead of scheduled maintenance of gas turbines has been proposed to avoid unnecessary shutdowns, save money and still protecting components from severe damage by continuous condition monitoring. Artificial neural networks could be used for this condition monitoring of gas turbines to decide the maintenance requirement. However, non-availability of operational data under varying conditions is a problem, especially for new gas turbines. Use of simulation data generated by the gas turbine manufacturer’s design program has been proposed to avoid this problem. To demonstrate this possibility, predictions by two ANN models, representing the same gas turbine, have been compared together with measured values from the plant. One model was trained with simulation data and the other one with operational data. This study has limitations as the two models were not developed for the basic objective of this study though used for this purpose, and apparently predictions by these two models were quite different. However, the predictions were almost equivalent when they were compared for identical conditions of model development. This shows that simulation data could be used for successful implementation of condition based maintenance of gas turbines through online condition monitoring. Furthermore, it is shown that no retraining of ANN models is needed after e.g. overhauls, a mere calibration is sufficient.

ACKNOWLEDGEMENTS

The authors are grateful for the financial support of the Swedish Gas Center AB (SGC) and Värme forsk AB. The authors are also grateful for the technical support from Siemens Industrial Turbomachinery AB.

REFERENCES


A novel approach for gas turbine monitoring combining CUSUM technique and artificial neural network
Fast M, Palmé T, Genrup M.
Presented at ASME Turbo Expo 2009, Orlando, Florida, USA
A NOVEL APPROACH FOR GAS TURBINE CONDITION MONITORING COMBINING CUSUM TECHNIQUE AND ARTIFICIAL NEURAL NETWORK

Magnus Fast
Division of Thermal Power Engineering
Department of Energy Sciences
Lund University
P.O. Box 118, S-221 00 Lund, Sweden

Thomas Palmé
Department of Mechanical and Structural Engineering and Materials Science
University of Stavanger
N-4036 Stavanger, Norway

Magnus Genrup
Division of Thermal Power Engineering
Department of Energy Sciences
Lund University
P.O. Box 118, S-221 00 Lund, Sweden

ABSTRACT
Investigation of a novel condition monitoring approach, combining artificial neural network (ANN) with a sequential analysis technique, has been reported in this paper. For this purpose operational data from a Siemens SGT600 gas turbine has been employed for the training of an ANN model. This ANN model is subsequently used for the prediction of performance parameters of the gas turbine. Simulated anomalies are introduced on two different sets of operational data, acquired one year apart, whereupon this data is compared with corresponding ANN predictions. The cumulative sum (CUSUM) technique is used to improve and facilitate the detection of such anomalies in the gas turbine’s performance. The results are promising, displaying fast detection of small changes and detection of changes even for a degraded gas turbine.

Keywords: ANN, CUSUM, condition monitoring, gas turbine

1 INTRODUCTION
A steadily increasing demand for effective and reliable gas turbine condition monitoring systems, as a means to increase the availability and reliability, has lately forced the original equipment manufacturers (OEMs) to, at least, offer such systems together with their engines. These systems are most often developed from the OEMs engine performance decks, which are equation based iterative solvers comprising engine component maps etc. However, there is a reluctance to disseminate source codes and maps at the same time as equation based iterative solvers might not be the best alternative for online tasks. Alternative approaches, using e.g., intelligent tools such as artificial neural network (ANN), have been investigated and evaluated for quite some time, with positive results. Researchers at the division of Thermal Power Engineering at Lund University have carried out several investigations [1-7] in this field and ANN has shown to be a good candidate for modeling of non-linear systems in the energy field and this work is a continuation of those previous studies.

ANN is strictly numerical and can be trained to recognize patterns using either operational data [4], from an actual engine, or simulation data, from an OEMs performance deck [7]. A trained ANN model is very small, due to its parallel structure, simple and consists only of straight forward equations, transferable into any programming language. The information is stored in the network’s weights, which consists of decimal numbers, and therefore no information about the gas turbine’s configuration is revealed. It is possible to achieve very high predictive performance with a properly configured and trained ANN using good quality data representing a broad range of operation. The predictions from an ANN model represents expected behavior of a gas turbine and can be continuously compared with corresponding measurements to detect any unexpected change, viz., condition monitoring. In this paper a technique is proposed to improve the detection of such changes by employing the CUSUM method. CUSUM
stands for cumulative sum and is a sequential analysis technique used for change detection purposes. By employing the CUSUM for change detection an ANN based condition monitoring tool can be better utilized through earlier detection of unwanted change in operation. Furthermore, an ANN model trained with operational data is tailored to a specific engine while an ANN model trained with simulation data is a more general representation of the engine type since all engines have their individualities. However, even when training an ANN with operational data, over time a level shift will appear between the predictions and measured values, due to degradation. This level shift, present from the beginning or developed over time, can complicate the change detection and would normally require a calibration or retraining of the ANN model. By employing the CUSUM method, calibration and retraining may prove unnecessary, since change detection is possible through studying directional changes of the cumulative sum instead of the actual values. In this paper an ANN is trained with operational data from a Siemens SGT600 gas turbine whereupon the proposed method is evaluated.

2 BRIEF BASICS OF ANN

ANN is a non-linear statistical data modeling tool that basically learns from experience (i.e., adaptive) [8]. Instead of being built a priori from specification, neural and adaptive systems use external data to automatically set their parameters [9]. ANN can be used to solve a variety of tasks, including classification, regression, general estimations problems, etc.

An ANN consists of a group of interconnected artificial neurons processing information in parallel. The performance of a network can be improved by rendering it “aware” of its output(s) through a performance feedback loop that includes a cost function. The feedback is used to adjust the network parameters through systematic procedures called learning or training rules, in order to improve the system output with respect to the desired goal [9].

The most common feedback algorithm is the back-propagation algorithm, proposed by Rumelhart et al. in 1986 [10], which provides a computationally efficient method for evaluation of the error function associated to each weight. The updating algorithm is based on a gradient descent method, where the updating algorithm updates the weights in order to minimize the error function. The multi-layer feed-forward network, more specifically the multi-layer perceptron (MLP), is the type of network which has been used during this study. It consists of an input layer, one or more hidden layer(s) and an output layer. However, there is no impediment to having more than one hidden layer, since it has been proven that one layer with hidden neurons is enough to approximate any continuous function if it only has a sufficient number of neurons [11]. Once the inputs are presented to the network they will be multiplied by their adjustable weights and in each processing element summed and passed through a transfer function in order to produce outputs. The data used as inputs is transmitted through the network, layer by layer, and a set of outputs is obtained. Errors are generated by comparing these outputs with the desired outputs. The errors are then used for updating the weights before another set of inputs is transmitted through the network, i.e., supervised training. The mean squared error (MSE) is calculated between each epoch (iteration) and the training is terminated when the MSE is satisfactory low. When the training is completed the weights are fixed and the network is ready to predict outputs from previously unseen data, i.e., generalization.

Before commencement of the training process the data set is randomized and divided into three subsets, training, cross-validation and test set. The training set is used by the network for weight adjustment (i.e., learning) and the cross-validation set is used to avoid overtraining by the network. The test set is used after the training to ensure that good generalization capability is obtained. A properly trained network displays similar prediction errors, or accuracy, in all three data sets.

Even though ANN is adaptive, the actual network configuration has to be specified. Selection of the number of hidden neurons is carried out by a trial- and error procedure where the network is trained multiple times with different number of neurons in the hidden layer. The network exhibiting the highest prediction accuracy, in all three data sets, is selected. The number of neurons needed in the hidden layer is determined by the complexity of the input/output parameter dimension. It is preferable to select as few neurons as possible, without sacrificing the prediction accuracy, to provide good generalization capabilities. Each neuron includes a transfer function, in which the incoming data is processed. In this study, based on accumulated experience at the division, the tangent hyperbolic function is used as transfer function.

3 BASIC PRINCIPLES OF THE CUSUM TECHNIQUE

The CUSUM technique was first proposed by Page [12] in 1954, for use in industrial process control to detect deviations in production parameters from pre-determined values. Before motivation of the CUSUM algorithm, the standard Shewhart control chart is introduced.

In the statistical theory, control charts are widely used to detect a shift in the mean value from a process, the most common are the Shewhart control charts [13]. The main goal with these charts is to distinguish natural variation from unnatural behavior. Several types of control charts exists, e.g., the “X chart” which is used for controlling central tendency and the “R chart” (or σ chart) which is used for controlling variability. If a variable is normally distributed, with the mean \( \bar{x} \) and the standard deviation \( \sigma \), the probability that \( x \) will fall within the interval \( [\bar{x} - 3\sigma, \bar{x} + 3\sigma] \) is 0.9973. If instead a subgroup of \( n \) observations is considered, the new interval will be \( [\bar{x} - 3\sigma/\sqrt{n}, \bar{x} + 3\sigma/\sqrt{n}] \). If a change between the expected and measured value occurs, the Shewhart charts provides a good detection tool. The major drawback of the Shewhart control charts is their inability to use the information contained in a sequence of plotted points. As a result, changes between the predicted mean and the measured mean are hard to detect if the change occurs inside the confidence limit set by the Shewhart chart.

The CUSUM technique was developed to solve the problem of detecting a change in a parameter, with emphasis on small changes. The main reason for detecting an unnatural behavior in a process is to provide a basis for action as soon as possible. In Engineering Statistics [13], two alternative types
of control charts are introduced with the purpose of using “past” knowledge. The first one is called signed sequential rank control chart (SSRCC) and the second cumulative sum control chart (CSCC), both of which have stopping rules that incorporate the information contained in all the sample points. This can be illustrated by considering a small shift in the mean value inside the upper and lower limits (the confidence limits) that may not be detected with the standard Shewhart charts. The CSCC is based on plotting the cumulative difference between the expected mean and the measured value for a parameter. The process is assumed to be in “control” as long as these points behave properly. Before the computational era, a detection chart called the V-mask was constructed in which failing data could be observed [13]. Good monitoring systems will take a long time before they indicate that an adjustment is necessary when the process is in control, i.e., an incorrect action, and take a short time to call for action when the process is no longer in control. This is measured by the average run length (ARL). A full explanation of how to calculate the ARL can be found in Engineering Statistics [13], which also provide a theoretical base for calculating the design parameters for the V-mask.

Several types of CUSUM algorithm are available and they all have their origin in the work proposed by Page [12]. In Detection of Abrupt Changes: Theory and Application [14] a description of the CUSUM algorithm used in offline and online applications is thoroughly discussed. In this study, an approach proposed by Lucas et al. [15] is used. This approach is fairly intuitive with few parameters to adjust for implementation. The first step is to compute the standardized deviations of observations from the desired process mean, shown in Eq. (1).

\[
z_i = \frac{x_i - \bar{x}}{\sigma_x}
\]

(1)

Where \(x_i\) is the observed value at time \(i\), \(\bar{x}\) is the desired process mean and \(\sigma\) is an estimation of the standard deviation of the observed values. These are accumulated over time to compute the cumulative sum, \(S\), at each time point \(i\) as seen in Eq. (2).

\[
S_i = S_{i-1} + z_i, \text{where } S_0 = 0
\]

(2)

If there is a shift in the process mean, \(z_i\) will tend to be larger or smaller than the target and the CUSUM will either steadily increase or decrease. Depending on the magnitude of the shift in the mean value, the CUSUM will not detect the change immediately and will require a number of observations at the new level before the change in the mean is recognized. In Fast Initial Response for CUSUM Quality-Control Schemes [15] the authors proposed a pair of cumulative sums (see Eq. (3)) where the first is used for detecting an increase in the mean, and the second for detecting a decrease in the mean, the so called two-sided CUSUM control chart.

\[
\begin{align*}
S_{HI} &= \max \{0, (z_i - k) + S_{HI-1} \} \\
S_{LI} &= \min \{0, (z_i + k) + S_{LI-1} \}
\end{align*}
\]

(3)

The parameter \(k\) is a reference value or allowable slack in the process. A usual choice of \(k\) is 0.5 and is an appropriate choice when a shift of one standard deviation is desired to be detected in the process mean. The CUSUM chart includes lower and upper alarm limits, or confidence limits, of the magnitude \(± h\sigma\) with \(h = 4\) or \(5\). These limits are used as a means of differentiating between random changes and sustained changes in the mean value. It should be recognized that a CUSUM chart does not immediately detect a change in the mean, it requires a run of values at the new mean value before it indicates that a change has occurred. Four different parameters need to be specified in order to implement the algorithm. These include the standard deviation, the “batch” size of the process mean, the slack parameter and the confidence limits.

4 CASE STUDY
The purpose of this case study is to investigate how the CUSUM technique can be advantageously employed, together with ANN, for the condition monitoring of a gas turbine. The investigation is twofold and the first step is to implement the CUSUM technique according to the theory explained in section 3. For this purpose operational data from a healthy and clean gas turbine, where ANN predictions and measurements match very closely, is employed. The second part of the study focuses on how the CUSUM technique can be employed if there is a level shift, between the ANN predictions and the measurements. For this purpose operational data is acquired from the same gas turbine after one year of operation. The second part is of certain interest since even if the monitoring tool, in this case an ANN model, have very high precision, over time a shift between this model and the actual machine will develop, due to degradation or component exchanges. It is not realistic to assume that the ANN model can be retrained every time this shift grows too large so the preferable scenario would be to have a monitoring system able to cope with this reality.

An already developed and validated ANN model, representing clean and healthy performance of a SGT600 gas turbine individual, will be employed for this study. The performance predictions are compared with corresponding measurements and through introducing small errors on these measurements the CUSUM technique is evaluated. More specifically, a Siemens SGT600 machine (described in 4.1) is modeled with ANN (described in 4.2) and the use of the CUSUM technique is evaluated by comparing ANN predictions with a healthy gas turbine (described in 4.3) and also with the same gas turbine after some time of operation (described in 4.4).

4.1 The SGT600 Gas Turbine
The power plant Gunnesboverket is owned by Lunds Energi AB and used for peak load operation during the winter months. It is a cogeneration plant consisting of a Siemens SGT600 gas turbine and a heat recovery unit connected to the district heating grid. The SGT600 gas turbine is a two-shaft unit and is thus suitable for both power generation and mechanical drive. The axial flow compressor has ten stages, of which the first two have a variable geometry, and generates a
pressure ratio of 14:1. The combustion chamber is of annular type and is built for dual fuels, i.e., gas and liquid. This particular gas turbine is always operated at full load, producing around 22 MW of electricity [16].

The gas turbine is also equipped with an anti-icing system to avoid harmful ice formation during simultaneous cold and humid conditions.

4.2 The ANN Model
The ANN model employed for the investigation in this study was previously developed for offline performance simulation and online condition monitoring of the SGT600 gas turbine at the power plant Gunnesboverket. The work was presented in the paper Development and multi-utility of an ANN model for an industrial gas turbine [4] where different aspects regarding data acquisition and neural network configuration and training were discussed. To summarize, the ANN model was constructed with the multi-layer feed-forward network type and trained with operational data using back-propagation. A baseline was established prior to the data acquisition and before the training the data was filtered to avoid the inclusion of outliers in the process. Also, with a sample frequency of one minute the data was considered unsuitable for modeling transient operation and data representing start ups and shut downs was removed. Consequently, the ANN model is only valid for the steady state operation of the gas turbine. The data was divided into three sets, one for training, one for cross-validation and another for independent testing after the training process. After the initial training a sensitivity analysis was conducted to verify the interdependence between the chosen input and output parameters, which resulted in a final network structure of 4H7 (i.e., four inputs, H hidden neurons and seven outputs). Finding the optimum number of hidden neurons was a trial and error process. The network structure, with input and output parameters is illustrated in Figure 1.

[Figure 1. ANN model configuration.]

The ambient conditions, i.e., temperature, pressure and relative humidity, all affect a gas turbine’s performance and are hence included as input parameters. The anti-icing switch adopts the discrete value ‘1’ or ‘0’, indicating whether the anti-icing system is turned on or off, thus assisting the ANN model in recognizing the two different modes of operation. Since the gas turbine always operates on full load no load parameter is needed. The output parameters were chosen based on real life needs as well as availability of reliable data. With this ANN model the performance of the SGT600 gas turbine at Gunnesboverket can be predicted with very high accuracy, shown in Table 1.

### Table 1. Error distribution for the ANN predictions [4].

<table>
<thead>
<tr>
<th>Error [%]</th>
<th>&lt;1</th>
<th>1-2</th>
<th>2-4</th>
<th>&gt;4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass flow rate of air</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Compressor outlet pressure</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Compressor outlet temperature</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mass flow rate of natural gas</td>
<td>6950</td>
<td>548</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Turbine outlet temperature</td>
<td>7498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Power output</td>
<td>7494</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Carbon dioxide emissions</td>
<td>7452</td>
<td>46</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

4.3 CUSUM Evaluation Part 1
The accuracy of the ANN model can be illustrated through plotting the ANN predictions together with corresponding measurements for a healthy machine, seen in Figure 2. The power output is the chosen parameter for this comparison and the sudden shifts in the parameter values are caused by the anti-icing system being turned on or off. Other, more continuous, variations of the power output are due to natural changes in the local ambient conditions. This plot represents about five days of operation, which corresponds to 7498 data points (one sample per minute). However, the data set is, in order to facilitate the visualization, compressed to 394 data points before being plotted. This compression is not result altering and merely consists of, in this particular case, plotting every 19th data point instead of every data point.

[Figure 2. Measurements and ANN predictions.]

In order to minimize the risk of encountering any unwanted influences during this investigation a portion of the data, without anti-icing in operation, is selected for further studies (i.e., data point 315 to 373 in Figure 2). This section represents approximately 19 hours (1131 data points) of operation and is plotted again in Figure 3 (compressed to 226 data points).
Traditionally, the CUSUM technique is employed to processes where data samples are normally distributed around a constant mean (as explained in section 3). In this case the constant mean will be represented by the ANN predictions and the distribution will be computed in terms of how far above or below the measurements are from these predictions. The standard deviation is calculated according to Eq. (4) and used to demonstrate the distribution of the measurements for the entire data set (7498 data points), illustrated in Figure 4.

$$
\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}
$$  (4)

As it appears the measurements are normally (or close to normally) distributed around its corresponding predictions, which is an indication that the ANN model is properly trained. It also signifies that the conditions for employing the CUSUM technique are favorable because the cumulative sum will remain around zero as long as the gas turbine is performing satisfactorily.

Figure 3. Measurements and ANN predictions.

Figure 4. Distribution of measurements.

From these control charts it can be concluded that both charts remain around zero, which indicates that the gas turbine behaves as expected.

To illustrate the potency of the CUSUM method a small continuing power drop of 15 kW is introduced, on the measurements, after data point 100 whereupon the equivalent plots are generated. Figure 6 shows the power output measurements and predictions, similar to Figure 3. Figure 7 shows the Shewhart and CUSUM control charts, similar to Figure 5.

Figure 5. Shewhart and CUSUM control charts.

Figure 6. Measurements and ANN predictions with a 15 kW power drop after data point 100.
The detection of such a small power drop is far from obvious when comparing the measurements and predictions in Figure 6 or looking at the Shewhart control chart in Figure 7. However, the power drop becomes very visible when studying the CUSUM control chart in Figure 7. In this example the detection of the 15 kW power drop takes in the order of 1½ hours using the CUSUM technique. The speed of detection is however dependant on many factors, such as update frequency, sample grouping size, alarm limits and value of the cumulative sum before the power drop occurs. For large enough deviations the Shewhart control chart would be sufficient but for fast detection of small deviations the CUSUM control chart is very powerful.

The strength of the CUSUM technique, regarding fast detection of small changes in operation, is hereby demonstrated. However, limitations arise almost immediately since all gas turbines are subject to performance deterioration during operation. A small permanent shift in a gas turbine’s performance will quickly cause the cumulative sum to diverge from zero. In this next section focus will be on circumventing this issue.

4.4 CUSUM Evaluation Part 2
This section aims at investigating how the CUSUM technique can be employed together with ANN for the detection of changes in operation even when there is a difference between the monitoring tool and the measurements.

Apparently, the CUSUM technique cannot be employed in the same manner as previously described since a continuing difference between measurements and predictions will cause the cumulative sum to rapidly diverge from zero. The approach is instead to consider the direction of the cumulative sum and changes in this direction. If the difference between measurements and predictions remain constant the cumulative sum will generate a somewhat straight line. When additional deviations occur this line will change direction. This is visualized in Figure 8 where simulated data is used for demonstration purposes.

In this figure the dots represent the computed cumulative sum and the dashed lines its different mean lines, indicating the direction of the cumulative sum. The starting point of the cumulative sum is at \( y \) equals zero and its steady decrease simulates a constant shift between measurements and predictions. At data point \( x \) an additional deviation is added which causes the cumulative sum to change direction. If \( j \) represents the present time index and \( x \) marks the start of the occurrence, then \( j-x \) equals the number of sample groups used for computing the new direction of the cumulative sum and \( x-i \) the number used for computing its initial direction. \( x \) is unknown and identifying it without generating false warnings, but still keeping a reasonable detection speed, is a matter of determining \( j-x \), \( x-i \) and \( \delta_{\text{max}} \), which is the maximum tolerable angle between the two mean lines.

To exemplify the approach a data set acquired one year after the ANN model was developed is employed. The period between the acquisitions of the data sets include an overhaul, with the installation of a new combustion chamber, and the accumulation of around 5000 operating hours, including approximately 40 start ups. Both the overhaul and the fact that several thousand operating hours have been accumulated are expected to alter the gas turbine’s performance. This is also illustrated in Figure 9 where the predictions from the ANN model are compared with corresponding measurements. As seen in the figure the measured power output is about 500 kW lower than what is predicted (expected), indicating substantial degradation.

To ensure the equitability of the comparison the “new” data set is collected for a period with similar operating conditions as before, i.e., similar ambient temperature, pressure, relative humidity and without the anti-icing system in operation. The plot in Figure 9 is also created using the same number of data points as the plot in Figure 3.
Although there is a shift between the measured and predicted power output, the behavior of the gas turbine seems to be well imbedded in the ANN model. This is verified through transferring the measurements to a secondary y-axis and displacing this axis to make the plots overlap, seen in Figure 10.

As an initial step in the assessment of the proposed method a 50 kW power drop is added after sample group 150 whereupon the cumulative sum is plotted (see Figure 11). 50 kW is chosen as it is a power drop substantial enough to be of interest to detect. The initial mean line is created through using the cumulative sums up till sample group 150 and subsequently the second mean line is created using the cumulative sums after sample group 150. As seen in the figure the 50 kW power drop generates a significant change in mean line direction. The angle, \( \delta \), in this particular case is on the order of four degrees, which gives an appraisal of what angles are acceptable and which are not. This example is based on the assumption that the starting point, \( x \), of the power drop is known. Furthermore, all the accumulated cumulative sums are used to calculate the mean lines.

<table>
<thead>
<tr>
<th>( j-x )</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>4.89°</td>
<td>4.62°</td>
<td>4.35°</td>
<td>4.05°</td>
</tr>
<tr>
<td>75</td>
<td>5.14°</td>
<td>4.61°</td>
<td>4.06°</td>
<td>3.52°</td>
</tr>
<tr>
<td>100</td>
<td>4.38°</td>
<td>3.59°</td>
<td>2.81°</td>
<td>3.01°</td>
</tr>
<tr>
<td>125</td>
<td>1.56°</td>
<td>3.34°</td>
<td>1.52°</td>
<td>3.49°</td>
</tr>
<tr>
<td>150</td>
<td>1.97°</td>
<td>3.87°</td>
<td>1.94°</td>
<td>3.97°</td>
</tr>
</tbody>
</table>

Also in this table, italic numbers indicate the maximum registered angle when a 50 kW power drop is added after sample group 150, as demonstrated in Figure 11. In the case where only one angle is presented no increase was registered compared to the “healthy” gas turbine. This shows the importance of choosing reasonable values for \( j-x \) and \( x-i \). To further clarify, if \( j-x \) is chosen to consist of 30 sample groups (i.e., two and a half hours) and \( x-i \) chosen to consist of 125 sample groups (i.e., ten and a half hours), then \( \delta \) could be expected to vary below 1.52 degrees. However, when introducing the error of 50 kW \( \delta \) reaches a maximum of 3.49 degrees. This means that if \( \delta_{max} \) were to be set at two degrees, the 50 kW power drop would have been detected. On the other hand, if \( j-x \) and \( x-i \) would be chosen too small, e.g., 30 and 75, the 50 kW power drop would not have caused a greater \( \delta \) than...
the normal operation and hence it would not be possible to detect.

With this section it is shown that through using the proposed method, of studying the direction of the cumulative sum instead of the actual values, detection of small changes in operation is possible as long as certain variables are chosen with care.

5 CONCLUSIONS AND DISCUSSION
Condition monitoring of a gas turbine through the combination of a generalized CUSUM technique and ANN has been proposed and demonstrated in this paper. The aim with the research is to address the difficulty of long term matching of a monitoring tool, e.g., an ANN model, with an actual engine. As mentioned in the beginning of the paper, an ANN trained with operational data is an excellent representation of that specific engine. However, this is only true for a short time span since the gas turbine’s performance will change over time, thus complicating the condition monitoring. The other scenario discussed is if an engine performance deck is used to produce data for the ANN modeling. Using simulation data, instead of operational data, is an attractive alternative since data can be produced in the exact resolutions and intervals needed. However, this type of ANN will be a more general representation of that engine type. So, regardless of what method is chosen, a difference between the ANN model and the actual engine will occur at some point in time. To avoid time consuming calibration- or retraining processes the proposed solution includes the employment of a modified CUSUM approach. This approach is based on the assumption that, despite any differences between an ANN model and an actual engine, the behavior of the engine is still embedded in the ANN model.

With this paper a new approach is presented and the results show that the proposed method is applicable and could be a possible solution to the problem. However, further investigations are needed to address the limitations of this study, such as how to handle different load levels, transient operation and false alarm rates.

ACKNOWLEDGMENT
The authors wish to acknowledge Dr. Agne Karlsson at Siemens Industrial Turbomachinery AB for his help and cooperation as well as the personnel at Lunds Energi AB. The Swedish Gas Center (SGC) is acknowledged for their financial support.

NOMENCLATURE

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>Number of hidden neurons</td>
</tr>
<tr>
<td>S</td>
<td>Cumulative sum</td>
</tr>
<tr>
<td>h</td>
<td>Confidence limit constant</td>
</tr>
<tr>
<td>k</td>
<td>Slack parameter</td>
</tr>
<tr>
<td>n</td>
<td>Subgroup</td>
</tr>
<tr>
<td>n</td>
<td>Number of samples</td>
</tr>
<tr>
<td>x</td>
<td>Value</td>
</tr>
<tr>
<td>z</td>
<td>Standardized deviation</td>
</tr>
<tr>
<td>δ</td>
<td>Angle</td>
</tr>
<tr>
<td>σ</td>
<td>Standard deviation</td>
</tr>
</tbody>
</table>

Subscripts

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>High</td>
</tr>
<tr>
<td>i</td>
<td>Number</td>
</tr>
<tr>
<td>L</td>
<td>Low</td>
</tr>
</tbody>
</table>

Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>Artificial neural network</td>
</tr>
<tr>
<td>CUSUM</td>
<td>Cumulative sum</td>
</tr>
<tr>
<td>OEM</td>
<td>Original equipment maker</td>
</tr>
</tbody>
</table>

REFERENCES
Gas turbine sensor validation through classification with artificial neural networks
Fast M, Palmé T, Karlsson A.
Presented at ECOS 2009, Foz do Iguaçu, Paraná, Brazil
GAS TURBINE SENSOR VALIDATION THROUGH CLASSIFICATION WITH ARTIFICIAL NEURAL NETWORKS

Magnus Fast, magnus.fast@energy.lth.se
Division of Thermal Power Engineering, Department of Energy Sciences, Lund University, P.O. Box 118, S-221 00 Lund, Sweden

Thomas Palmé, thomas.palme@uis.no
Department of Mechanical- and Structural Engineering and Materials Science, University of Stavanger, N-4036 Stavanger, Norway

Agne Karlsson, agne.karlsson@siemens.com
Siemens Industrial Turbomachinery AB, S-612 83 Finspång, Sweden

Abstract. Modern power plants are all strongly dependent on reliable and accurate sensor readings for their monitoring and control, which makes the sensors an important part of any plant. Failing sensors can force a plant or component into non-optimal operation, failing sensors can also cause complete shut-downs of operation or in the worst case scenario cause damage to components. Due to their importance sensors need regular calibration and maintenance which is a time-consuming and therefore costly process. In this paper a method for evaluating sensor accuracy is presented with the goal of minimizing the need for calibration and at the same time avoiding shut-downs due to sensor faults etc. The proposed method consist of training artificial neural networks as classifiers to recognize sensor drifts. The method is evaluated on two types of gas turbines, i.e., one single-shaft and one twin-shaft machine, and the results show early detection of sensor drifts for both types of machines as well as accurate production of soft measurements.

Keywords: sensor validation, gas turbine, classification, artificial neural network

1. INTRODUCTION

The deregulation of the electricity market, and the consequent increase in competitiveness, drives the power producers to continuously investigate various means of keeping/increasing their profits. Improving the electrical efficiency through hardware upgrades is probably the most commonly employed measure, although the interest for improvements in plant utilization is on the rise. Improvement in plant utilization is often measured in reliability, availability and maintainability (RAM). Reliability and availability are measures of how many hours the plant operation can be upheld without unplanned or planned stops respectively. If one considers that the outage time is 3.6 days per percent reliability, then the need for efficient measures is clear. Especially, since many units of today have requirements of above 99 percent reliability. By improving the monitoring of a power plant or its components, there is a contingency of increasing the RAM. Improved monitoring could decrease the time spent on unplanned stops (i.e., increase the reliability) at the same time as maintenance could become more condition based instead of time based, which would extend the periods between overhauls (thereby increasing the availability).

Modern power plants are depending on accurate and reliable sensor readings for the control and monitoring of the plant and are therefore equipped with a considerable number of sensors. To avoid potential erroneous sensor signals affecting the operation of a plant, redundant sensors are often employed for crucial measurements. Sensors need regular maintenance and calibration, regardless of their condition, and the greater the number of sensors the more costly and time consuming this procedure becomes. Through employing some method of sensor validation, operational failures due to sensor faults can be avoided at the same time as sensor calibration can be based on sensor’s condition rather than schedules.

Sensor validation relies upon correlation between parameters and two common methods are principal component analysis (PCA) and partial least squares (PLS), both of which are linear methods, i.e., optimal when applied to linear systems. Another option is Kalman filters, which also are developed for linear systems but can be extended to cope with nonlinear systems. However, Kalman filters are model based (i.e., not data driven) and cannot be constructed with e.g., operational data. The success of Kalman filter models for sensor validation is therefore dependent on the fidelity of the system or component model. Artificial neural networks (ANNs) are one candidate which are able to tackle nonlinear systems as well as being developed from data without the need of model specifications (i.e., data driven). In 1991, Kramer proposed ANN for performing data compression by feature extraction without sacrificing the nonlinearity in the data (Kramer, 1991) and in Autoassociative neural network (Kramer, 1992), he extended his evaluation of the method and puts it into a sensor validation context and names it autoassociative neural network (AANN). This network is an identity mapper, containing three hidden layer layers with the following function in order: mapping, data compression and de-mapping. The actual implementation of an AANN for sensor validation is not straight forward and different approaches are suggested for the different sensor validation steps that are fault detection, isolation and accommodation (FDIA). One approach is to train the AANN with a healthy data set and subsequently evaluate the differences between inputs and outputs in order to detect a failing sensor. One problem with this method is that a failing sensor value is
actually an extrapolation by the network which may cause poor performance. Another approach, also proposed by Kramer, is to train the AANN in two stages where in the first stage the AANN is trained with healthy data. In the second stage the weights in all layers except the layer connected to the input nodes, i.e., the mapping layer, are locked whereupon the network is trained again, this time with failing sensor values, and the mapping layer is optimized with regards to its number of neurons. This method is called robust AANN. In Sensor validation and fault detection using neural networks (Xu et al., 1999), the use of AANN is evaluated for a fossil power generation plant, and the AANN is also complemented with a statistical decision algorithm, the so called sequential probability ratio test (SPRT), to isolate failing sensors. In (Mattern et al., 1998) and (Moller et al., 1998), robust AANN are developed for aircraft gas turbines and both studies implements the two stage training procedure mentioned above. In (Moller et al., 1998), combinations of dual sensor faults were tested with mixed results for different combinations. It should also be highlighted that in both (Mattern et al., 1998) and (Moller et al., 1998) the main goal was to produce accurate sensor estimates before the gas turbine’s control algorithm in order to avoid catastrophic failure of the flight vehicle due to sensor failure. Hence, an immediate correction of a sensor failure is crucial. This is achieved by the AANN since the forward pass through the network consists of only one equation. In (Ogaji et al., 2002), AANN is evaluated for both single and dual sensor failure scenarios. Here, the AANNs are trained with both healthy and faulty data at the same time, which differs from (Mattern et al., 1998) and (Moller et al., 1998) where a two stage training of the AANN is performed.

In this paper an alternative approach, employing a classification neural network trained with simulated sensor faults, is evaluated. The goal with this approach is to achieve earlier detection of faulty sensors, than with the traditional AANN approach, and producing more accurate recovered values. In this study, single sensor faults are considered, even though the method may be extended to multiple sensor faults. For the evaluation of this proposed method two types of gas turbines, one single-shaft and one twin-shaft, will be considered. Data representing gas turbine parameters, such as pressures and temperatures, is produced with a Siemens performance deck for a selected range of operation. Simulated sensor faults are added to this data, together with different levels of white noise, whereupon a classification neural network is trained to detect and isolate failing sensors. Finding the minimum detectable level of sensor fault for respective parameters is a trial-and-error process. Once these levels are established the final classification network is evaluated with regards to generalization through employing an independent test data set. Also, separate neural networks are trained for the production of recovered values once a failing sensor has been identified. These networks utilize the healthy sensor values to produce a soft measurement of the failing sensor.

1.1. Single- and twin-shaft gas turbines

The industrial gas turbines are typically one-shaft or two-shaft units. The choice of the number of shafts (i.e., single or multiple) is dependent on the intended application. In a single-shaft gas turbine arrangement, the generator is connected to the same shaft as the compressor and the turbine, which renders the components mechanically dependent. Single-shaft gas turbines are mostly suited for fixed-speed operation, such as base-load power generation. In a two-shaft layout, the turbine is split into one part that drives the compressor and one that operates the load. The compressor, combustion chamber and compressor turbine is generally referred to as a gas generator, and the turbine driving the load is called a power turbine. The gas generator and power turbine are hence mechanically independent and can be operated at different rotational speeds. In other words, the gas generator can be run at maximum speed, resulting in a good efficiency while the power turbine runs at the speed of the load. (Walsh & Fletcher, 1998; Razak, 2007)

The incitement of studying both types of gas turbines is to extend the evaluation of the proposed method for sensor validation. The two types of machines differ significantly from one another, in terms of control and off-design behavior, which is illustrated with Fig. 1 and Fig. 2 below. These figures correspond to Siemens SGT600 and SGT800 gas turbine models, which are chosen to be representative of the single- and twin-shaft gas turbine types.

Figure 1. Operational behavior of a twin-shaft gas turbine (SGT600)
A single-shaft gas turbine’s power level is controlled by the air flow, through different VIGV settings. Initially when the engine is started, the VIGV’s are fully closed to maintain compressor stability. At some load point, when the maximum exhaust temperature is reached, the engine control is taken over by the VIGV setting and the firing controller maintains either maximum exhaust temperature or the nominal combustor outlet temperature (whichever one is reached first). This control strategy is beneficial for both combined cycle performance and CO/UHC emissions. A high exhaust temperature is beneficial for the steam cycle performance, hence providing relief for the reduced gas turbine efficiency at part load. The high firing temperature has a positive impact on the CO/UHC emission since it is a strong function of the firing temperature. A twin-shaft unit is quite different in terms of controlling, and there are no practical means at hand to control the engine mass flow. Hence, the only available mean is to control the firing temperature and let the mass flow drop with the engine matching. These two different behaviors are reflected in Fig. 1 and Fig. 2 above.

2. SENSOR VALIDATION THROUGH CLASSIFICATION WITH ARTIFICIAL NEURAL NETWORKS

Multi layer perceptrons (MLPs) have universal mapping capabilities and Cybenko showed in 1989 that a one hidden layer MLP is able to approximate any input and output relationship given enough number of neurons in the hidden layer (Cybenko, 1989). In a regression network, the network is trained to approximate the underlying relationships between input and output parameters. In a classification network, the goal is rather different. The classification network needs to create decision surfaces which minimize the classification error. Traditional classification methods are normally parametric, which means that the discriminant function has a well defined mathematical form, e.g., Gaussian that depends on a set of parameters, mean and variance. The performance of a parametric classifier depends on the choice of the likelihood model, which has to be predetermined. Neural networks on the other hand are considered to be semi-parametric, which means that the data is used to create the discriminant functions that separate different classes. Furthermore, the adjustable weights in a neural network scale linearly with the dimension while the number of parameters to be adjusted increases much faster for a parametric classifier with increasing dimensionality. A MLP configured with one hidden layer and nonlinear transfer functions in the hidden layer is able to create any arbitrary discriminant function. The capability to train hidden layers is made possible by the back-propagation algorithm which is a reformulation of the chain rule of calculus. The back-propagation algorithm was first invented by Werbos (Werbos, 1974) but popularized by Rumelhart (Rumelhart et al., 1986). When the MLPs are used in a classification task, the goal is to classify different input patterns into different classes. The class assignment is mutually exclusive and hence need a non-linear mechanism, such as all or nothing. Both the classification and the regression problems seek systems that transform inputs into desired responses but the details of the mappings are different. In the classification task, the network is trained to minimize the classification error while in the regression network the network is trained to minimize the difference between input and output values. The classification network can be trained with the mean squared error (MSE) cost function which will cause the network to produce a decision surface between the classes that separates them equally in between. For this to work the classes should be equally big and the batch training method employed, i.e., the network weights are updated after seeing all data patterns. To create an optimal classification MLP it has to be assumed that the hidden layer has a sufficient number of neurons to produce the required map from input space to output targets. Also, it has to be assumed that the training data is sufficient and that the training takes the learning system to the global minima.

2.1. Data preparation and neural network training

The classification neural network is configured in following manner: Each state is divided into one class where e.g., the state “healthy” is one class containing only healthy sensor readings. Each sensor, which corresponds to one parameter, such as compressor outlet pressure, is divided into two classes, one class for positive sensor drift and one class for negative sensor drift. This approach has two advantages, the first one being that information about the direction of the drift is provided and the second being that the neural network classification capability improves
compared to the case where both positive and negative sensor drifts are combined in one class. Consequently, the final data set grows depending on the number of parameters and the number of data patterns in the healthy data set according to Eq. (1).

\[ dp_f = dp_h \cdot p \cdot 2 + dp_h \cdot 2 \]  

(1)

where \( dp_f \) is the final number of data patterns, \( dp_h \) is the number of data patterns in the data set containing healthy data and \( p \) is number of parameters (sensors). Table 1 illustrates this data preparation procedure based on two sensors, where the top two rows correspond to the healthy measurements and the succeeding rows correspond to 10% drift in sensor 1 and 20% drift in sensor 2 respectively.

<table>
<thead>
<tr>
<th>Sensor 1 reading</th>
<th>Sensor 2 reading</th>
<th>Class 1: Healthy</th>
<th>Class 2: S1 too high</th>
<th>Class 3: S2 too high</th>
<th>Class 4: S1 too low</th>
<th>Class 5: S2 too low</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.1</td>
<td>2.4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.5</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2.4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>7.2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.9</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4.5</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1.6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>4.8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The classification network is configured with one hidden layer and trained with an enhanced back-propagation learning rule called the momentum rule. This learning algorithm belongs to the class of heuristic techniques of improving the standard back-propagation algorithm based on gradient descent. The momentum learning algorithm is based on the standard gradient descent method but extended with a momentum term which provides a filtering effect on the learning trajectory and helps the network to pass local minima without being trapped in them. The momentum learning algorithm requires three parameters to be set, namely:

- Step size in the output layer
- Step size in the hidden layer
- Momentum term in both weight layers

The step size is increased backwards in the network since the back-propagation algorithm decreases the error associated to each weight when passing from one layer to another. By increasing the step size backward in the network the learning rate becomes similar in all layers and saturated neurons (i.e., neurons without learning capability) are avoided.

2.2 Producing recovered values

The MLP networks developed to accommodate failing sensor values are configured as regression networks, where the healthy sensor values are used to predict the true value from the failing sensor (so called soft sensoring). The configuration is a one hidden layer MLP, with a nonlinear transfer function in the hidden layer and a linear transfer function in the output layer. In these networks, the training algorithm is represented by the scaled conjugate gradient (SCG) algorithm. The SCG algorithm is an implementation of the standard conjugate gradient (CG) algorithm which is an improvement of the gradient descent learning algorithm. Instead of evaluating the local approximation (i.e., the derivative) of the cost function CG algorithms evaluate the curvature and hence it converges faster. A detailed description of the SCG algorithm can be found in *A scaled conjugate gradient algorithm for fast supervised learning* (Møller, 1993). One of the advantages of this algorithm is that no parameters are predetermined before training.

3. GAS TURBINE SENSOR VALIDATION

The suggested method, of using ANN as a classifier for sensor validation purposes, is evaluated on two types of gas turbines, one single-shaft and one twin-shaft machine. As described in the Introduction these two types of machines differ significantly from one another, in terms of control and off-design behavior, and through evaluating the suggested method on both types the results are substantiated. The data used for ANN modeling and evaluation is produced, for a
broad range of operation, with an allocated performance deck license from Siemens, which ensures accurate and highly realistic data. Also, a level of white noise has been added to this data to further resemble real conditions and to assess the method’s sensitivity towards noise. The data production, ANN structuring and modeling have been kept similar, for the two types of machines, to the furthest extent for the equitability of the study. The detection of a faulty sensor and the production of a recovered value are thought to be performed before the faulty value reaches the gas turbine’s control system, therefore avoiding the faulty sensor reading affecting the operation of the gas turbine.

3.1. Parameter selection and network structuring

The parameters are chosen based on certain guidelines; they should be measured in reality, they have to be of significance and they have to be available in the simulation program. Also, the parameters are, as far as possible, chosen equal for both gas turbines. The chosen parameters, consisting mostly of temperatures and pressures, for the SGT800 are illustrated in Fig. 3.

![Figure 3. Schematic representation of an SGT800 with selected measurements](image)

Since each parameter is accompanied with two classes, one for positive drift and another for negative drift, together with one class representing the healthy state of all the sensors the number of outputs from the networks will be twice the number of inputs plus one. The neural network structures for the two sensor validation models are shown in Fig. 4. Some exceptions are made compared to the guidelines for parameter selection, e.g., the mass flow rates included in the models (compressor- and fuel mass flow rate) are not directly measured in reality but calculated based on other measurements, such as the bellmouth pressure drop. This is an adjustment based on availability of desired parameters in the simulation program.

![Figure 4. Neural network structure for the two gas turbine sensor validation models, SGT600 (left) and SGT800 (right)](image)

3.2. Data generation and preparation

For this study simulation data has been chosen instead of operational data in order to generate optimal conditions for evaluating the proposed method for sensor validation. Through this selection the data can be produced in desirable intervals and resolutions, suitable for ANN training, and unwanted influences from a real machine can be eliminated. The simulation data is produced for varying loads and ambient temperatures, to simulate a broad range of operation, using Siemens “in-house” performance deck, which is an equation based iterative solver comprising engine component
maps etc. The relative humidity and ambient pressure are kept constant at ISO conditions (i.e., 60 % and 1.013 bar respectively), as well as the fuel’s composition and heating value. The load is varied from 10 % to 100 % in steps of five percent and the ambient temperature is varied from -25°C to 25°C in steps of five degrees, which results in a total of 209 operational combinations. These data patterns correspond to the healthy state of the sensors and are the foundation for the ANN training data set. The operational variation for each parameter can be seen in Table 2.

Table 2. Operational intervals for selected parameters and gas turbines

<table>
<thead>
<tr>
<th>x</th>
<th>SGT600</th>
<th>SGT800</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>P&lt;sub&gt;gen&lt;/sub&gt;</td>
<td>2.20 – 27.8 MW</td>
<td>4.40 – 53.5 MW</td>
<td></td>
</tr>
<tr>
<td>t&lt;sub&gt;amb&lt;/sub&gt;</td>
<td>± 25 °C</td>
<td>± 25 °C</td>
<td></td>
</tr>
<tr>
<td>t&lt;sub&gt;c&lt;/sub&gt;</td>
<td>41.3 – 88.2 kg/s</td>
<td>- kg/s</td>
<td></td>
</tr>
<tr>
<td>ΔP&lt;sub&gt;bell&lt;/sub&gt;</td>
<td>- bar</td>
<td>0.0023 – 0.0061 bar</td>
<td></td>
</tr>
<tr>
<td>IGV</td>
<td>- °</td>
<td>- 40 – 1.2 °</td>
<td></td>
</tr>
<tr>
<td>t&lt;sub&gt;j&lt;/sub&gt;</td>
<td>182.9 – 394.0 °C</td>
<td>317.8 – 451.8 °C</td>
<td></td>
</tr>
<tr>
<td>P&lt;sub&gt;3&lt;/sub&gt;</td>
<td>5.90 – 15.8 bar</td>
<td>10.7 – 22.2 bar</td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;f&lt;/sub&gt;</td>
<td>0.4 – 1.8 kg/s</td>
<td>0.9 – 3.0 kg/s</td>
<td></td>
</tr>
<tr>
<td>t&lt;sub&gt;7&lt;/sub&gt;</td>
<td>273.9 – 553.0 °C</td>
<td>366.3 – 604.7 °C</td>
<td></td>
</tr>
<tr>
<td>P&lt;sub&gt;7&lt;/sub&gt;</td>
<td>1.013 – 1.022 bar</td>
<td>1.065 – 1.152 bar</td>
<td></td>
</tr>
</tbody>
</table>

As described in section 2.1, faulty data is produced, and for each defined class another 209 data patterns are generated, which results in a significant increase in the total number of data patterns. To verify the trained ANNs generalization capability a separate test data set is also produced parallel to the training data set but for different operational combinations. More specifically, the test data set is produced through varying the load from 12.5 % to 97.5 % in steps of five percent and the ambient temperature from -22.5°C to 22.5°C in steps of five degrees. This is exemplified in Fig. 5 where both the training data points and the test data points, corresponding to a healthy sensor, for the exhaust gas temperature of the SGT800 gas turbine are plotted.

![Figure 5. Training- and test data points for healthy measurements of the exhaust gas temperature for the SGT800 gas turbine](image)

To emulate real conditions different levels of white noise have been added to each parameter. The levels of white noise, indicated in Tab. 3, are chosen so that they correspond to those of real sensors in a gas turbine. The noise is generated with a randomization algorithm within the given intervals, thus resulting in it being normally distributed.
Table 3. Measurement uncertainties used for white noise production

<table>
<thead>
<tr>
<th>X</th>
<th>$\tau_x$ (at full load)</th>
<th>$\tau_x$, SGT600</th>
<th>$\tau_x$, SGT800</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{\text{gen}}$</td>
<td>$\pm 0.2$ MW</td>
<td>$\pm 0.72$ %</td>
<td>$\pm 0.37$ %</td>
</tr>
<tr>
<td>$t_{\text{amb}}$</td>
<td>$\pm 0.1$ °C</td>
<td>$\pm 0.40$ %</td>
<td>$\pm 0.40$ %</td>
</tr>
<tr>
<td>$m_{c}$</td>
<td>$\pm 0.08$ kg/s</td>
<td>$\pm 0.09$ %</td>
<td>-</td>
</tr>
<tr>
<td>$\Delta p_{\text{bell}}$</td>
<td>$\pm 0.00015$ bar</td>
<td>-</td>
<td>$\pm 2.47$ %</td>
</tr>
<tr>
<td>IGV</td>
<td>$\pm 0.2$ °</td>
<td>-</td>
<td>$\pm 0.20$ %</td>
</tr>
<tr>
<td>$t_{3}$</td>
<td>$\pm 0.1$ °C</td>
<td>$\pm 0.03$ %</td>
<td>$\pm 0.02$ %</td>
</tr>
<tr>
<td>$p_{3}$</td>
<td>$\pm 0.02$ bar</td>
<td>$\pm 0.13$ %</td>
<td>$\pm 0.09$ %</td>
</tr>
<tr>
<td>$m_{r}$</td>
<td>$\pm 0.02$ kg/s</td>
<td>$\pm 1.11$ %</td>
<td>$\pm 0.66$ %</td>
</tr>
<tr>
<td>$t_{7}$</td>
<td>$\pm 2$ °C</td>
<td>$\pm 0.36$ %</td>
<td>$\pm 0.33$ %</td>
</tr>
<tr>
<td>$p_{7}$</td>
<td>$\pm 0.001$ bar</td>
<td>$\pm 0.10$ %</td>
<td>$\pm 0.09$ %</td>
</tr>
</tbody>
</table>

3.3. Neural network training and performance

The process of finding the best neural network configuration for a certain application almost always requires some level of trial-and-error, and this case is no exception. In addition to optimizing the network itself, with regards to number of hidden neurons, data normalization limits etc., the smallest drift level for each parameter has to be identified through, for each combination of drifts, training new neural networks. Basically, one chooses drift levels for each parameter, trains a network and evaluates its classification capacity, whereupon the initial drift levels are adjusted and new networks trained until satisfactory results are achieved. Table 4 shows the results of four such optimizations, two for each type of gas turbine, where the first network (for each gas turbine type) is optimized without any noise in the data and subsequently the second network is optimized with white noise in the data. The table also shows, apart from the smallest detectable drifts, the upper/lower limits of validity for classification of drifts in each parameter. This means that when a sensor drift grows too large the network can exhibit problems of correctly classifying it. However, these limits are so far away from the actual healthy parameter value so that it will not cause any problems. The limits given in Tab. 4 have been evaluated on both the training- and the test data (described in section 3.2) to ensure the networks generalization capability, with the requirement that an average of 99 % is correctly classified. This means that sensor drifts, within given intervals, will be detected (i.e., properly classified) ninety-nine times out of a hundred (most sensor drifts have a correct classification ratio of 100 % within their given intervals).

Table 4. Drift detection intervals with an average correct classification ratio of over 99 %

<table>
<thead>
<tr>
<th>Class</th>
<th>SGT600 w/o noise</th>
<th>SGT600 w noise</th>
<th>SGT600 w/o noise</th>
<th>SGT600 w noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (H)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2 ($P_{\text{gen}}$)</td>
<td>0.9 % max</td>
<td>2.0 % max</td>
<td>2 ($P_{\text{gen}}$)</td>
<td>0.9 % max</td>
</tr>
<tr>
<td>3 ($t_{\text{amb}}$)</td>
<td>0.8 % max</td>
<td>1.3 % max</td>
<td>3 ($t_{\text{amb}}$)</td>
<td>1.0 % max</td>
</tr>
<tr>
<td>4 ($m_{c}$)</td>
<td>0.4 % max</td>
<td>0.5 % max</td>
<td>4 ($\Delta p_{\text{bell}}$)</td>
<td>0.5 % max</td>
</tr>
<tr>
<td>5 ($t_{3}$)</td>
<td>0.5 % max</td>
<td>0.7 % max</td>
<td>5 ($IGV$)</td>
<td>0.5 % max</td>
</tr>
<tr>
<td>6 ($p_{3}$)</td>
<td>0.3 % max</td>
<td>0.7 % max</td>
<td>6 ($t_{3}$)</td>
<td>0.3 % max</td>
</tr>
<tr>
<td>7 ($m_{r}$)</td>
<td>0.5 % max</td>
<td>2.5 % max</td>
<td>7 ($p_{3}$)</td>
<td>0.4 % max</td>
</tr>
<tr>
<td>8 ($t_{7}$)</td>
<td>0.5 % max</td>
<td>1.0 % max</td>
<td>8 ($m_{r}$)</td>
<td>0.5 % max</td>
</tr>
<tr>
<td>9 ($p_{7}$)</td>
<td>0.1 % max</td>
<td>0.2 % max</td>
<td>9 ($t_{7}$)</td>
<td>0.5 % max</td>
</tr>
<tr>
<td>10 ($P_{\text{gen}}$)</td>
<td>-0.9 % -15 %</td>
<td>-2.0 % -32 %</td>
<td>10 ($p_{7}$)</td>
<td>0.1 % max</td>
</tr>
<tr>
<td>11 ($t_{\text{amb}}$)</td>
<td>-0.8 % -13 %</td>
<td>-1.3 % -15 %</td>
<td>11 ($P_{\text{gen}}$)</td>
<td>-0.9 % -14 %</td>
</tr>
<tr>
<td>12 ($m_{c}$)</td>
<td>-0.4 % -10 %</td>
<td>-0.5 % -13 %</td>
<td>12 ($t_{\text{amb}}$)</td>
<td>-1.0 % -12 %</td>
</tr>
<tr>
<td>13 ($t_{3}$)</td>
<td>-0.5 % -13 %</td>
<td>-0.7 % -25 %</td>
<td>13 ($\Delta p_{\text{bell}}$)</td>
<td>-0.5 % -18 %</td>
</tr>
<tr>
<td>14 ($p_{3}$)</td>
<td>-0.3 % -8 %</td>
<td>-0.7 % -35 %</td>
<td>14 ($IGV$)</td>
<td>-0.5 % -10 %</td>
</tr>
<tr>
<td>15 ($m_{r}$)</td>
<td>-0.5 % -8 %</td>
<td>-2.5 % min</td>
<td>15 ($t_{3}$)</td>
<td>-0.3 % -13 %</td>
</tr>
<tr>
<td>16 ($t_{7}$)</td>
<td>-0.5 % -8 %</td>
<td>-1.0 % -16 %</td>
<td>16 ($p_{3}$)</td>
<td>-0.4 % -9 %</td>
</tr>
<tr>
<td>17 ($p_{7}$)</td>
<td>-0.1 % min</td>
<td>-0.2 % min</td>
<td>17 ($m_{r}$)</td>
<td>-0.5 % -11 %</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>18 ($t_{7}$)</td>
<td>-0.5 % -8 %</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>19 ($p_{7}$)</td>
<td>-0.1 % min</td>
</tr>
</tbody>
</table>

To exemplify, class six of the second SGT600 model (trained and evaluated with noisy data) is chosen from Tab. 4 and corresponds to positive drift of the compressor outlet pressure sensor (see Fig. 4). Any positive drift in this sensor, between 0.7 % up to the parameter’s operational limit will be detected by the classification network. Here, 0.7 per cent corresponds to 0.11 bars at full load or 0.04 bars at the lowest load. Noticeable from Tab. 4 is that in order to reach the
same correct classification ratio when training with noisy data the drift levels for training has to be increased somewhat. On the other hand, this has a positive side-effect as well since the ANNs trained with slightly higher drift levels have wider drift detection intervals. Another aspect noticeable from Tab. 4 is that all positive sensor drifts are classified correctly within the entire operational range while the majority of the negative sensor drifts have slightly more narrow drift detection intervals.

Since the simulation data is produced by varying two parameters, load and ambient temperature, it is possible to illustrate the classes as regions in a three dimensional space and thereby visualizing the accuracy of the evaluated classification method. An example is made using the exhaust gas temperature of the SGT600 gas turbine and plotting classes eight (positive drift) and sixteen (negative drift), shown in Fig. 6. Actual parameter values, drift levels and intervals are displayed in the figure.

![Figure 6. Representation of two classes in a three dimensional space for positive and negative drift of an SGT600 exhaust gas temperature sensor](image)

As long as the parameter value (i.e., the exhaust gas temperature measurement) remains in between the two classes, represented by the regions in Fig. 6, the sensor is deemed as healthy. Once the parameter value crosses into one of these regions it will be classified as drifting, positive drift for the upper region and negative drift for the lower region. Positive sensor drift can be detected in the entire operational range, as for all parameters, while negative drift has a lower limitation at 16% of the correct value. However, this lower level is large enough to be fully detectable with conventional methods, such as logical expressions.

### 3.4 Producing recovered values

Once a failing sensor has been identified by a classification neural network a recovered value can be reproduced through using the remaining, healthy sensors. This is achieved through employing individual regression neural networks, using the healthy sensors as inputs and the desired parameter as output, which is an easy and reliable way of producing accurate recovered values. Table 5 shows the prediction accuracy of the ANN soft sensing models where the average error is calculated through using the predictions from the ANN models, which is trained with noisy data, and comparing it with corresponding values without the noise (i.e., the actual parameter values). A closer look at Tab. 5 reveals that most of the sensor values are reproduced below the given levels of white noise, indicating that the noise is “filtered” out during the training of the networks. This also means that the ANN predictions are at least as good as those of a healthy sensor.
Table 5. Average error for prediction of soft measurements

<table>
<thead>
<tr>
<th>$x$</th>
<th>SGT600</th>
<th>SGT800</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{amb}$</td>
<td>0.03%</td>
<td>0.23%</td>
</tr>
<tr>
<td>$P_{gen}$</td>
<td>0.24%</td>
<td>0.33%</td>
</tr>
<tr>
<td>$m_c$</td>
<td>0.07%</td>
<td>-</td>
</tr>
<tr>
<td>$\Delta p_{bell}$</td>
<td>-</td>
<td>0.39%</td>
</tr>
<tr>
<td>IGV</td>
<td>-</td>
<td>0.11%</td>
</tr>
<tr>
<td>$t_3$</td>
<td>0.08%</td>
<td>0.08%</td>
</tr>
<tr>
<td>$p_2$</td>
<td>0.07%</td>
<td>0.07%</td>
</tr>
<tr>
<td>$m_f$</td>
<td>0.35%</td>
<td>0.23%</td>
</tr>
<tr>
<td>$t_7$</td>
<td>0.10%</td>
<td>0.23%</td>
</tr>
<tr>
<td>$p_2$</td>
<td>0.02%</td>
<td>0.03%</td>
</tr>
</tbody>
</table>

4. SUMMARY AND DISCUSSION

This study has demonstrated a complete solution for sensor fault detection, isolation and accommodation through employing ANN as a classifier in combination with ANNs configured as a regression networks for the production of soft measurements. An important advantage to highlight is that the algorithm can be developed with operational data without the need of a detailed model of the system. The method has been tested on two types of gas turbines, one single-shaft machine and one twin-shaft machine, and good results, in terms of detection of small sensor drifts as well as production of accurate soft measurements, have been achieved in both cases. By evaluating two different gas turbine types the method’s generic capability is highlighted. However, the method is applicable not only on the specific machines evaluated in this study but on all types of systems and components where sensors have some type of internal correlation. In this study single sensor faults have been tested and one natural extension of this study would be to evaluate multiple sensor faults. Other interesting extensions of this study would be to assess the influence of component degradation, and combining the method with a sequential analysis technique, such as cumulative sum (CUSUM), to achieve even earlier drift detection through studying sequences of data.

As often when dealing with data driven methods and neural networks there is a lot of trial-and-error work involved, which makes it impossible to exactly state what can be expected when applying this method. Detection of sensor faults depends on many factors, such as the amount of data used for training, what resolution it has and in what operational range it is available, the levels of noise in the data, the number of parameters considered etc. One of the main issues with this method is to select a well representative data set representing the healthy state, since the final data set grows substantially which results in long training periods when considering optimization of e.g., the number of neurons in the hidden layer.

ACKNOWLEDGEMENTS

The authors of this paper greatly acknowledge Siemens Industrial Turbomachinery AB for providing the necessary tools for this study as well as assisting with technical support. The authors also express their gratitude towards the Swedish Gas Center AB (SGC) for their financial support.

NOMENCLATURE

<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Auto-associative neural network</th>
</tr>
</thead>
<tbody>
<tr>
<td>AANN</td>
<td>Artificial neural network</td>
</tr>
<tr>
<td>ANN</td>
<td>Conjugate gradient</td>
</tr>
<tr>
<td>CG</td>
<td>Carbon monoxide</td>
</tr>
<tr>
<td>CO</td>
<td>Combustor outlet temperature</td>
</tr>
<tr>
<td>COT</td>
<td>Cumulative sum</td>
</tr>
<tr>
<td>CUSUM</td>
<td>Fault detection, isolation and accommodation</td>
</tr>
<tr>
<td>FDIA</td>
<td></td>
</tr>
<tr>
<td>Subscripts</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Class</td>
</tr>
<tr>
<td>H</td>
<td>Healthy</td>
</tr>
<tr>
<td>P</td>
<td>Power</td>
</tr>
<tr>
<td>S</td>
<td>Sensor</td>
</tr>
<tr>
<td>dp</td>
<td>Pressure difference, also data pattern</td>
</tr>
<tr>
<td>m</td>
<td>Mass flow rate</td>
</tr>
<tr>
<td>p</td>
<td>Pressure, also parameter</td>
</tr>
<tr>
<td>t</td>
<td>Temperature</td>
</tr>
<tr>
<td>$x$</td>
<td>Parameter</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Noise level</td>
</tr>
</tbody>
</table>

Greek

Abbreviations

| $\tau$ | Noise level |

Subscripts

| $3$ | Compressor outlet |

Diagram
REFERENCES

Werbos, P. J., 1974, "Beyond regression: New tools for prediction and analysis in the behavioral sciences": Harvard University.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.