Decision Trees and Genetic Algorithms for Condition Monitoring Forecasting of Aircraft Air Conditioning

M. Gerdes
Philotech GmbH, Brauereiweg 4, D - 21614 Buxtehude

16th December 2013

Unscheduled maintenance of aircraft can cause significant costs. The machine needs to be repaired before it can operate again. Thus it is desirable to have concepts and methods to prevent unscheduled maintenance. This paper proposes a method for forecasting the condition of aircraftt air conditioning system based on observed past data. Forecasting is done in a point by point way, by iterating the algorithm. The proposed method uses decision trees to find and learn patterns in past data and use these patterns to select the best forecasting method to forecast future data points. Forecasting a data point is based on selecting the best applicable approximation method. The selection is done by calculating different features/attributes of the time series and then evaluating the decision tree. A genetic algorithm is used to find the best feature set for the given problem to increase the forecasting performance. The experiments show a good forecasting ability even when the function is disturbed by noise.

1 Introduction

Unscheduled maintenance costs are a significant cost factor in aircraft operation[1]. Aircraft operators have significant increased costs, if an aircraft departure is delayed or canceled. One source of unscheduled maintenance is when a part of an aircraft needs to be replaced or repaired before the scheduled replacement time. The aircraft air conditioning and filtering system is such a system. The air filters clog faster or slower depending on the environment conditions where the aircraft is mainly operating. Filters clog faster in a moister environment and slower in a dry environment. A clogged filter system may not only cause a delay but also passenger discomfort. Currently the air condition system is monitored by using pressure sensors to detect changes in the air pressure. Forecasts are done by comparing the measurement against an empirical curve, which relates the pressure difference to the probability of clogging and a forecast for the mean time to clogging [2].

This paper describes a method to use machine learning to forecast the condition of a system. The method uses a decision tree to decide what the best method to forecast a future data point is and a genetic algorithm to adapt the decision tree to the current problem to improve the performance. The decision of the best forecasting method is based on learned patterns from past data. Motivation for this approach was to have a simple method to forecast the condition of the air conditioning system, which can adapt itself to different time series based on the operation of the aircraft and handle the influence of events on time series. The method should be easy to understand by an operator and should be able to adapt itself to different problems without much need for human interaction/expert. A time series of the system condition may be constant or linear for a long time, but suddenly an event happens and the time series changes significantly. Forecasting such a time series is difficult. The use of a decision tree enables the proposed method to use the best available forecasting method based on learned experience to adapt better to the new condition. An advantage of this method is, that it can use currently existing sensors and forecasting concepts to calculate the forecast. A genetic algorithm is used to increase the performance of the forecasting by searching for the optimal features set which generates the best decision tree for the given problem.

1.1 Time Series

A time series is a chronological sequence of observations on a particular variable[3]. This can be the production of a company, the DOW, a temperature, a pressure difference or a system condition. The history of a system condition can be seen as a single or multi dimensional time series. If the condition of a system is represented only by a single variable then the resulting time series is a one dimensional time series. If the condition is represented by two or more different variables then the resulting time series is a multidimensional time series. Prediction of future events and conditions is called forecast, the act of making such a prediction is called forecasting[3]. Common methods for time series forecasting are:

- Simple linear regression[4]
- Polynomial Regression[3]
- Multi Regression[4]
- Moving average[4]
- Exponential smoothing[4]
- Autoregressive integrated moving average (ARIMA)[4]

Marin Goluband Andrea Budin Posavec propose the use of genetic algorithms for adapting the approximation functions for forecasting[5], Animesh Chaturvedi and Samanvaya Chandra use quantitative data and a neural network to forecast financial time series[6].

1.2 Decision Trees

Decision trees are a method from the area of artificial intelligence and are used for machine learning[7]. They are often binary trees, where each node has an if-then-else function on an attribute of the sample data. More complex versions with more than two branches use a switch function. Training of the tree can be done with the ID3 algorithm (Iterative Dichotomiser 3, published by J. Ross Quinlan in 1986[8]). ID3 was the first algorithm to construct decision trees. ID3 had some problems and was improved. The improved version of ID3 is C4.5[9]. There are other algorithms to construct a decision trees available, including random trees. Decision trees are easy to understand and a decision/classification can be calculated fast. A sample decision tree can be seen in Fig. 1.

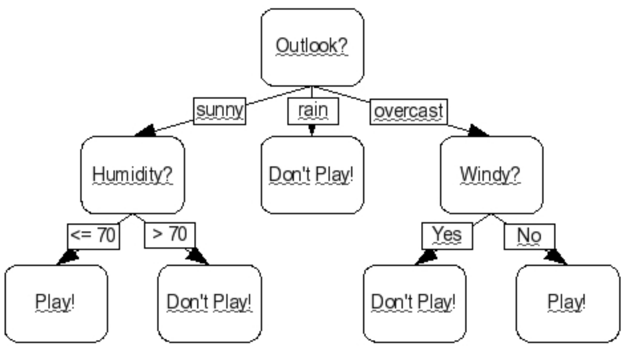

Figure 1: A simple decision tree

1.3 Genetic Algorithms

Genetic algorithms belong to the class of heuristic local search algorithms. They evaluate multiple valid solutions (population), choose (selection) the best (fitness) solutions and create new variations of those by combining (crossover) and changing (mutation) the solution. The new set of solutions is now evaluated and the best ones are selected again and combined and changed. Each iteration of these steps is called generation. The search is finished when the algorithm calculated a certain number of generations or when an abort criteria is reached[7].

2 Method

The method proposed in this paper for time series forecasting is based on decision trees. The inputs to a decision tree are time series characteristics (e.g. maximum value, gradient) and the output is an approximation function/method for forecasting based on training data. Forecasting quality is increased by using a genetic algorithm[7] for optimizing the process parameters. Optimization of the process parameters allows the process also to adapt itself to different problems without human interaction. Training enables the forecasting process to use experience of past data to predict the future data points in a much more reliable way then without training. This is obtained due to the fact that the process can learn when to use a different forecasting function then the obvious, because of irregularities in the time series. These irregularities can be triggered by the occurrence of certain events, that change the future data points of the time series significantly e.g. switching from a simple linear behavior to an exponential behavior.

The process is divided into two parts, one part for training of the algorithm and optimizing the decision tree and one part for forecasting the time series. In the training part training samples are created, time series features are calculated, a forecasting method is selected and the decision tree is generated. Data points are forecasted, after the decision tree is generated and the process parameters are optimized. Each iteration of the forecasting process calculates a single future data point. With multiple iterations it is possible to calculate more data points. Variations of the default process can calculate multiple data points and are shown later in this section.

2.1 Training Process

The learning process takes much more time than the forecasting process and is only executed when new training samples are available and during the initial training. Goal of the training process is to find an ideal set of features of the times series that give the most information for finding the optimal extrapolation algorithm. Input to the training process is a data set with different features and the best extrapolation algorithm for the time series that the features represent. The learning process does have seven steps. The first four steps are executed only once to generate the input for the last three steps. All steps except the last one (process parameter optimization) are iterated multiple times to generate a random base population of decision trees for the parameter optimization with a genetic algorithm (last step).

Samples Decision trees and most other concepts from artificial intelligence need many data samples for learning and finding patterns. For a time series this means, that a time series should not be to short and/or that multiple time series are available. Sample time series should include all relevant conditions and events. The algorithm can only learn from past data and thus it cannot predict events that were not in the sample data.

Process Parameters The training process is controlled by multiple parameters. These parameters control how samples and time series characteristics are calculated. Process parameters are:

- Window size [numeric]. This parameter defines how many data points each data sample contains. These data points include past data and the data points to forecast. This can be a fixed or a varying number, which is different for each data sample.

- Window shift [numeric]. This parameter defines by how many data points the sampling window should be shifted to generate a new data sample from the time series. Window shift and window size define how many training samples are generated.

- Forecast horizon [numeric]. The parameter controls how large the forecasting horizon is. This means for how many future data points the forecasting method should be calculated. The forecasting horizon can be from one data point up to all remaining data points in the time series. In the remaining document is a forecasting horizon of one data point used. Forecasting horizon cannot be larger than window size.

- Short Term Gradient calculation [Boolean]. This parameter defines, if the gradient for the data points in the current window shall be calculated.

- Long Gradient calculation [Boolean]. This parameter defines, if the gradient for the complete time series until the last data points shall be calculated.

- Mean value calculation [Boolean]. This parameter defines if the mean of the sample data points shall be calculated.

- Maximum value calculation [Boolean]. This parameter defines if the maximum of the sample data points shall be calculated.

- Minimum value calculation [Boolean]. This parameter defines if the minimum of the sample data points shall be calculated.

- Dimensions used for classification [Boolean list]. This parameter defines which dimensions shall also be used for characteristics calculation.

- Zero Crossing[Boolean]. This parameter decides if the number of zero crossings in the current window shall be calculated.

Other parameters may also be used. Including long term trends and short term trends is also possible by calculating the parameters for the current window and for the complete time series. The optimization algorithm then decides what long term and what short term parameters deliver the best results.

Time Series Features In this step the training samples are generated and time series features are calculated, based on the process parameters. First the time series is split into multiple smaller time series. The length of these smaller time series is defined by the window size. Data sample generation is done by shifting a n-data point window over the sample time series. See Fig. 2 for an example with a one point window shift and a window size of three. Other features may be used, depending on the problem and the need.

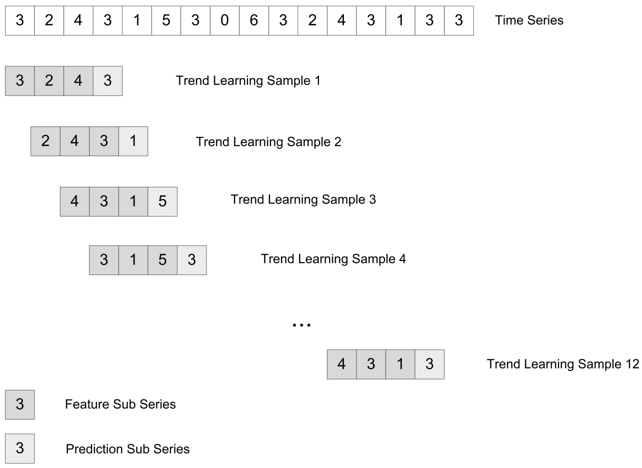

Figure 2: Generation of trend learning samples with a sliding window

Training Samples Next the decision tree output for the training time series needs to be calculated. For each possible approximation function/method from a given list is the forecasting for the forecasting horizon calculated. Next is the squared forecasting error calculated. The best fitting approximation function/method (with the least forecasting error) defines the "class" of the data sample. The following example is a list of possible simple approximation functions (simple linear regression, polynomial regression and multi regression), each approximation function can contain up to four parameters (a, b, c and d):

$$
\begin{aligned}
f(x) &= a \cdot x + br \\
f(x) &= a \cdot x^2 + b \cdot x + c \\
f(x) &= a \cdot x^3 + b \cdot x^2 + c \cdot x + d \\
f(x) &= a \cdot e^{b \cdot (x+c)} + d \\
f(x) &= a \cdot log(b \cdot (x + c)) + d \\
f(x) &= a \cdot sin(b \cdot (x + c)) + d \\
f(x) &= a \cdot cos(b \cdot (x + c)) + d \\
f(x) &= 1 - a \cdot e^{b*(x+c)} \\
f(x) &= a \cdot b^{x+c} \\
f(x) &= a \cdot (b \cdot (x + c))^2 + d
\end{aligned}
\tag{1}
$$

The given list was used in the later experiments. In addition to the previous list of simple functions it is possible to use other forecasting methods. Moving average, auto regressive process, moving average process, ARMA, ARIMA, exponential smoothing ... can also be used as a function. [10] did some experiments for using complex functions to forecast the complete future time series instead of only one future data point.

Decision Tree Training The decision tree is trained after all samples have been created using any decision tree learning algorithm like ID3, C4.5, random trees ...

Performance After the decision tree has been calculated, it can be evaluated and tested on the sample time series. The training data samples are applied to the forecasting process to calculate the rest of the sample time series from each training sample. Next the performance is calculated by calculating the maximum squared error of the forecasting and use this value as the fitness of the decision tree. Different approaches for calculating the fitness of the decision tree like the maximum confidence range are possible. The confidence range describes how many data points in a row can be forecasted until the error (either relative or absolute) is greater than a given limit.

Process Parameter Optimization If the best performance of the generated decision trees is below a given limit, then a genetic algorithm is used to generate new valid solutions. New decision trees are generated until one the decision trees does have a performance above the given limit or until a certain number of generations have been calculated. Input to the genetic algorithm are the process parameters and the fitness is measured based on the performance of the decision tree. This does have the advantage that the genetic algorithm adapts the process parameters for the decision tree to the current problem by selecting the time series features which give the best performance. In addition this also reduces the need for a human expert to set the process parameters.

2.2 Forecasting Process

The forecasting process is an iterative process. During each interaction a new data point is forecasted. The forecasting process is simple and consist of the following steps:

Time Series Features The first step in the forecasting process is to calculate time series features based on the optimized process parameters and given past data points.

Decision Tree Evaluation Next step is to evaluate the time series features by using the generated decision tree. The result is an approximation function.

Next Data Point(s) Prediction The next step is to use the approximation function to extrapolate the next data point of the time series.

Add Result to Time Series The new calculated data point is added at the end of the given time series. All such calculated data points are the forecasting.

Iterate If more than one point shall be forecasted then the forecasting process is repeated with the new added data point as part of the past data points. Each iteration step calculates a new data point.

2.3 Method Summary

The method consist of two processes: one training process and one forecasting process. Both processes use the same given set of methods to forecast the next data point for a given time series. Input to both processes are short (5-20 data points) time series snippets. The time series snippets contain one additional point for the training. In the training process is a decision tree calculated that is than used for the forecasting. The decision tree was calculated using process parameters that were optimized by a genetic algorithm. The forecasting is done point by point. The decision tree is used to select the best forecasting method for the current given time series snippet.

2.4 Process Modifications

The method can be modified in different ways based on the problem at hand. One modification is to not only use the past data points plus one future point for calculating the approximation function, but to use more than one future data point to account for a long term trend (greater forecast horizon[4]). The forecasting would still calculate only one future data point, but the long term trend is considered for generating the decision tree.

Another modification would be to iterate the training process only once and then use the calculated forecasting method to forecast all remaining future data points of the time series. This modification works well when the training data not only includes one future data point, but multiple future data points up to all remaining data points of the sample time series. [10] shows this modification.

A regression tree[11] can be used, if a future data point shall be directly forecasted without a approximation function. This reduces the needed processing power, but the process needs to iterate to forecast multiple data points.

3 Experiments

To validate the process three experiments have been performed. All experiments used a time series that is typical for the change of a system condition over time[12]. The first experiment was about forecasting the time series without any noise. In the second experiment noise was added to the test samples, while the training samples were without noise. For the third experiment noise was added to training and testing samples to get a more realistic use case. The forecasting of future data points was started at three different points in the time series: 1/4, 2/4 and 3/4 of the function. The time series was forecasted until the end of the sample time series and the number of past data points were ten with a forecasting horizon of one.

The time series for the experiment was a one dimensional time series that contained 120 data points. The first 80 data points were calculated via $f(x) = \frac{0.1}{80} \cdot x$ the data points between 81 and 120 were calculated by the following equation $f(x) = (\frac{x-81}{120-81})^2 + 0.1$. Fig. 3 shows the time series.

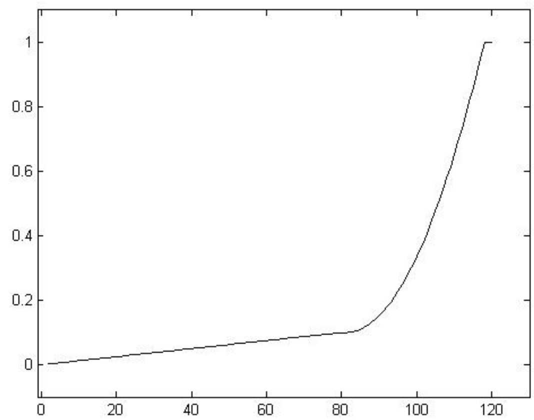

Figure 3: Plot of the experiment function

The quality of the forecasting was measured by six values. For every forecasting the maximum, mean and minimum forecasting error was calculated and the confidence range for an absolute error of 0.01, 0.05 and 0.1 was also calculated. The used genetic algorithm for the experiments had population of 50 members and 20 generations. New members were generated by dividing the process parameters into three parts and generating a child from three different parents. The mutation rate for each process parameter was 10 %.

3.1 Forecasting Without Noise

The first experiment was about forecasting the time series by using the method as it was presented. There was no noise in either training or forecasting data. In Fig. 4 the results of the forecasting are plotted. The black line are the input points, the red line is the sample time series to predict, while the green line is the calculated forecasting, starting at data point 30, 60 or 90. The image shall show the advantage of using a decision tree. The image shows on the right side the forecasting using the best prediction method without using a decision tree to select the method. This is the same method that is used to generate training samples in the training process. On the left side is the same input but the proposed process is used to forecast the next data point. It is visible that the advantage of the decision tree comes, when the function suddenly changes (data point 80) 0-degree polynomial to a second-degree polynomial function. Here is the decision tree forecasting method much more accurate than the simple method that uses only the best fitting method.

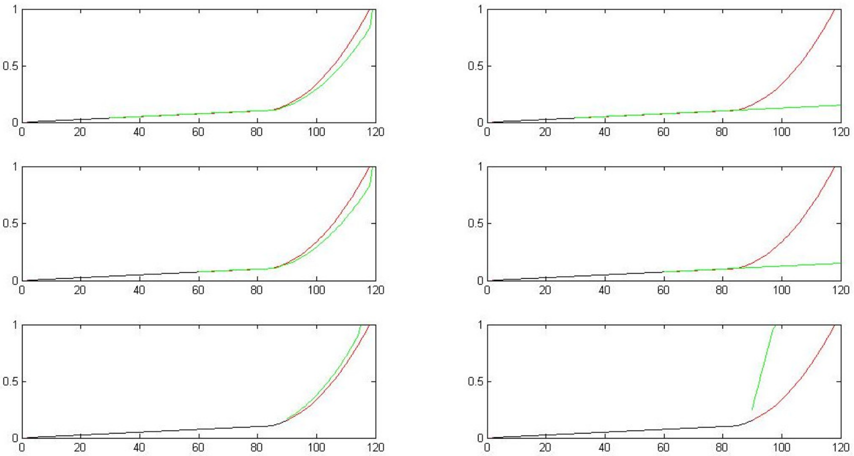

Figure 4: Plot with different starting points and forecast

Fig. 4 shows that the algorithm was quite good at following the time series. The maximum error was low and the algorithm was able to predict the time series nearly perfect. The preconditions for this forecasting were ideal (no noise) and the time series was pretty simple. The advantage of using a decision tree is clearly visible.

3.2 Forecasting with noisy test samples

In the second experiment the training samples were not changed, but the testing samples were modified by noise. A random number between -0.02 and 0.02 (2 % noise) or for

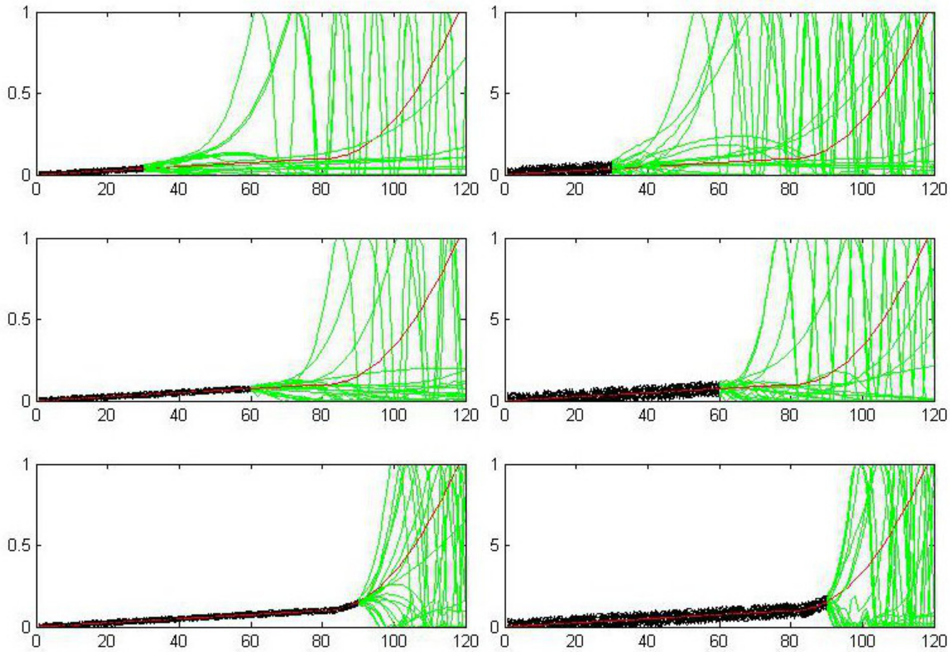

Figure 5: Plot with different starting points, noisy training data and noise (left: 2 % and right: 5 %)

the second test -0.05 to 0.05 (5 % noise) was added to each data point. Goal of the experiment was to check how well the method can work with noisy input data. Results are shown in Fig. 5. The test was repeated 20 times to show the forecasting results for different noisy input data.

Oscillations are caused by the used functions for the predictions like sin and polynomial functions. The forecasting accuracy was lower than without noise. If the starting point of the forecasting was at data point 60 then the results were especially bad. The algorithm was unsure at which position in the time series it was. The reason for this was the high impact of the noise on the time series features due to the relative small number of past data points (10). This let the decision tree make the wrong decision.

3.3 Forecasting noisy training and test samples

In this experiment the training data and testing data were noisy. Two tests with different noise levels were calculated. For each test three noisy training time series were used to generate training samples for decision tree calculation. Each data point in the time series was modified by a random number. This number was white noise between -0.02 and 0.02 (2 % noise) for the first test, -0.05 to 0.05 (5 % noise) in the second test. The results are shown in Fig. 6. The test was repeated 20 times to show the forecasting results for different noisy input data.

In the picture (Fig. 6) it is visible, that the accuracy of the forecasting is not so well,

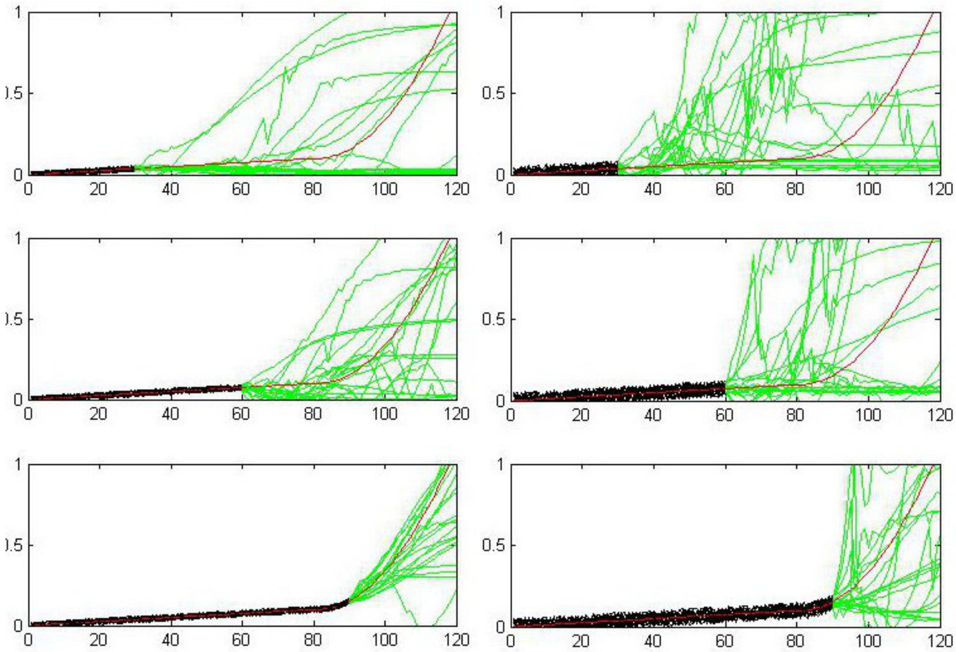

Figure 6: Plot with different starting points, noisy training and testing data and noise (left: 2 % and right: 5 %)

but this was to be expected. It is worth to note that only three noisy training time series were used. Still the predicted time series are similar to the original curve. The inaccuracy is created by the problem that the algorithm was not sure at which position it was and that the approximation only had ten noise points to calculate the next data point. With more training data and more past data points the algorithm will be able to perform better. With noise training and testing data the algorithm used different functions for the approximation and thus created no oscillations.

4 Conclusion

Experiments showed that the method is well suited for time series forecasting tasks. The advantage of using a decision tree is clearly visible and the performance of the forecasting is significantly improved. With more advanced forecasting methods it is possible to increase the performance. The method is able to adapt to different problems and the performance can be increased by using problem specific approximation functions/methods and process parameter optimization. A single future data point is forecasted (by offering a good approximation function) and then the process is iterated until a desired number of data points were forecasted. It is easily possible to train the algorithm to use an approximation function to forecast more than one data point if the problem desires this (long term forecasting). Forecasting quality of the method increases with the number of available past data points, more available features and more test and training samples especially when the training data is very noisy. The method can be enhanced by different concepts to have an increased quality of the forecast.

References

[1] M. Gerdes, D. Scholz, B. Randerath,"Reducing Delays Caused by Unscheduled Maintenance and Cabin Reconfiguration" *2nd International Workshop on Aircraft System Technologies, AST 2009* TUHH, Hamburg, 26./27. Mrz 2009. Aachen: Shaker, 2009, S. 109 - 119.

[2] Weber, Kirsten; Airbus Deutschland, EDYVVC: *Filter Clogging Indication of Recirculation Air Filters.* Confidential presentation, Feb. 2008.

[3] B. Bowerman, R. O'Connell, *Forecasting and Time Series : An Applied Approach* Duxbury Press: Belmont, California, 1993.

[4] D. Montgomery, L. Johnson, J. Gardiner, *Forecasting and Time Series Analysis 2nd edition*McGraw-Hill: New York, 1990.

[5] M. Golub, A. B. Posavec, *Using Genetic Algorithms For Adapting Approximation Functions* 19th International Conference on INFORMATION TECHNOLOGY INTERFACES '97, Pula, Croatia, 1997.

[6] A. Chaturvedi, S. Chandra, *A Neural Stock Price Predictor Using Quantitative Data* iiWAS 2004 - The sixth International Conference on Information Integrationand Web-based Applications Services, Jakarta, Indonesia, 2004.

[7] S. Russel, P. Norvig, *Artificial Intelligence: A modern Approach* Prentice Hall, 2. Auflage, 2003.

[8] J. R. Quinlan, *Induction of Decision Trees* Springer Netherlands, 1986.

[9] J. R. Quinlan, *Programs for Machine Learning* Morgan Kaufmann, 1993.

[10] A. Kret, *Statistische Analyse und Vorhersage von Zeitreihen zur vorausschauenden Wartung von Flugzeugkomponenten* Technical University of Hamburg-Harburg, Institute of Computer Technology, 2011.

[11] L. Breiman, J. Friedman, R. Olshen, C. Stone, *Classifcation and Regression Trees* Wadsworth: Belmont, CA, 1984.

[12] J. Kolerus, J. Wassermann, *Zustandsberwachung von Maschinen* expert : Renningen, 2008.