A synchronous filter for gear vibration monitoring using computational intelligence

by

Lungile Mndileki Zanoxolo Mdlazi

A dissertation submitted in partial fulfilment of
the requirements for degree

Master of Engineering

in the Department of Mechanical and Aeronautical Engineering
of the Faculty of Engineering, Built Environment and Information Technology

of the

University of Pretoria

March 2004

A synchronous filter for gear vibration monitoring using computational intelligence

by

Lungile Mndileki Zanoxolo Mdlazi

Supervisor	:	Prof. P.S. Heyns
Co-supervisors	:	Prof. T. Marwala
		Mr C.J. Stander
Department	:	Mechanical and Aeronautical Engineering
Degree	:	M.Eng.

Summary

Interaction of various components in rotating machinery like gearboxes may generate excitation forces at various frequencies. These frequencies may sometimes overlap with the frequencies of the forces generated by other components in the system. Conventional vibration spectrum analysis does not attenuate noise and spectral frequency band overlapping, which in many applications masks the changes in the structural response caused by the deterioration of certain components in the machine.

This problem is overcome by the use of time domain averaging (synchronous averaging). In time domain averaging, the vibration signal is sampled at a frequency that is synchronised with the rotation of the gear of interest and the samples obtained for each singular position of the gear are ensemble-averaged. When sufficient averages are taken, all the vibration from the gearbox, which is asynchronous with the vibration of the gear, is attenuated. The resulting time synchronously averaged signal obtained through this process indicates the vibration produced during one rotation of the monitored gear. This direct time domain averaging process essentially acts as a broadband noise synchronous filter, which filters out the frequency content that is asynchronous with the vibration of the gear of interest provided that enough averages

are taken. The time domain averaging procedure requires an enormous amount of vibration data to execute, making it very difficult to develop online gearbox condition monitoring systems that make use of time domain averaging to enhance their diagnostic capabilities since data acquisition and analysis cannot be done simultaneously.

The objective of this research was to develop a more efficient way for calculating the time domain average of a gear vibration signal. A study of Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs) was conducted to assess their suitability for use in time domain averaging. Two time domain averaging models that use ANNs and SVMs were developed. Model 1 uses a single feedforward network configuration to map the input which are rotation synchronised gear vibration signals to the output which is the time domain average of the gear vibration signal, using only a section of the input space. Model 2 operates in two stages. In the first stage, it uses a feedforward network to predict the instantaneous time domain average of the gear vibration after 10 inputs (10 rotation synchronised gear vibration signals) to predict the instantaneous average of the gear rotation. The outputs from the first stage are used as inputs to the second stage, where a second feedforward network is used to predict the time domain average of the entire vibration signal.

When ANNs and SVMs were implemented, the results indicated that the amount of gear vibration data that is required to calculate the time domain average using Model 1 can be reduced by 75 percent and the amount of gear vibration data that needs to be stored in the data acquisition system when Model 2 is used can be reduced by 83 percent.

Keywords: Artificial Neural Networks, Time Domain Averaging, Synchronous average, Multi-layer, Perceptron, Radial Basis Function, Support Vector Machines, Gearbox and Vibration

Acknowledgements

I would like to give a sincere word of thanks to the following people:

- My supervisors for their guidance and support throughout the period of this research.
  - Prof. P.S Heyns
  - Prof. T. Marwala
  - C.J. Stander

- Prof. P.S Heyns and C.J. Stander for development of the accelerated gear life test rig.

- J.G. Davel for the design of the gears used in this study.

- Mr F. Windell for his assistance with the experimental work.

- To the NRF and Eskom for financial support.

- My family and friends for their love and support throughout this period.

- To God Almighty for offering me the opportunity to study.

This work was done with financial aid from the DoL SCARCE SKILLS SCHOLARSHIP and the Eskom TESP fund.

Table of Contents

Chapter 1 – Introduction and literature survey

Chapter 2 – Existing time domain averaging models

Chapter 3 – Artificial neural networks and support vector machines

Appendix D

Nomenclature

Symbol	Description
$\|x - x^n\|$	Euclidian distance between **x** and $\mathbf{x}^n$
$\mathbf{\Phi}$	Square matrix with elements $\phi_{nn} = \phi(\|x - x^n\|)$
$\mathbf{\Phi}^{\dagger}$	Pseudo inverse
$\xi_i^- \xi_i^+$	Slack parameter presenting the lower and upper constraints
$R[f]$	Risk function
$R_{emp}[f]$	Empirical risk minimisation function
$K(x, x')$	Kernel functions
$(x)_+$	The positive part of x
O	Column vector of zeros
L	Loss function
α	Weight decay coefficient in MLP
α, α^*	Lagrange multipliers
σ	Variance
$\phi(\cdot)$	Basis function
$\upsilon(t)$	Rectangular window of unit amplitude
$\phi 0$	Extra basis function with activation fixed to 1
η_{sim}	Simulation accuracy
*	Convolution
$a(t)$	Revised window model time domain average
$A(t)$	Fourier transform of $a(t)$
C	SVM tolerance parameter
$c(t)$	Impulse signal
$C(t)$	Fourier transform of impulse signal
D	Training data set

E	Cost function
$e(t)$	Noise signal
e_{sim}	Response error
f	Frequency
f_f	Frequency of trigger signal
f_{inner}	Inner activation function
$f_{\max}$	Maximum frequency
f_{outer}	Outer activation function
f_s	Sampling frequency
$g(t)$	Deterministic periodic function of period T
$h(x^n)$	Interpolation functions
k	Number of gear rotations
M	Number of hidden units
N	Number of impulses in impulse train
n	Index for training pattern in MLP
n_t	Residual signal that is time locked to period T
$r(t)$	Infinite train of impulses
$R(t)$	Fourier transform of infinite train of impulses
t	Time
T	Period
t_k	Training target
$\mathbf{t}^n$	Target consisting of N input vectors
T_R	Period of rectangular window
T_s	Period between the sample points
T_t	Period of impulse signal
$V(f)$	Fourier transform of rectangular window
$\mathbf{W}$	Matrix of weights (w_n)
$w_{j0}^{(1)}$	Bias for hidden unit j
$w_{ji}^{(1)}$	First layer weights
w_{k0}	Radial basis function biases
w_{kj}	Weight parameters
w_{kj}	Basis function weights

X	Input to neural network
$\mathbf{X}$	Input space
$x(t)$	Time signal
$X(t)$	Fourier transform of time signal
X_k	Training input
$X_{\max}$	Maximum value of vibration during a given interval
x_n	Repetitive component of noise signal
$\mathbf{x}^n$	Data set consisting of N input vectors
x_p	Periodic component of signal
x_r	Continuous random process
Y	Output of neural network
$y(t)$	Time domain average
$y_{achieved}$	Obtained output
$y_{desired}$	Desired output
y_k	Output of MLP network
$z(t)$	Numerically generated time signal

Abbreviations

CG	Conjugate gradient
DSP	Digital signal processing
KKT	Karush-Kuhn-Tucker condition
MLP	Multi-layer peceptron
RBF	Radial basis function
RMS	Root mean square
RMSE	Root mean square error
SB	Side band frequency
SCG	Scaled conjugate gradient
SVMs	Support vector machines
TDA	Time domain average

Chapter 1

Introduction and literature survey

1.1 Introduction

Gears and gearbox systems are vital components in many industrial mechanical applications. A gearbox failure in a large mechanical system could easily lead to production losses. Early detection of incipient failure in gearboxes is, therefore, of great practical and commercial importance. It permits the plant operators and maintenance personnel to schedule shutdown and repair of the gearbox instead of unscheduled catastrophic failure.

Different signal processing techniques have been employed by operators and engineers to gather information about the condition of gearboxes to schedule maintenance activities. These techniques include oil debris analysis, vibration analysis, visual inspection and various non-destructive testing techniques. In recent years, neural networks have been used with much success in pattern recognition and fault identification (Bishop, 1995; Zhong et al., 2003; Fidêncio et al., 2003). Vibration based analysis has been used with success in detection of damage in structures and rotating machinery. Vibration-monitoring techniques are based on the assumption that changes in the measured structural response can be linked to the deterioration in the condition of the structure (McFadden, 1987). Recent advances in integrated circuit technology and digital signal processing has allowed for real time analysis of vibration response, in both the frequency and the time domain to be performed. If permanent transducers can be mounted on the structures, online condition monitoring of the vibrating structure could be possible, resulting in a much safer working environment.

This is however not necessarily true when monitoring incipient failure on rotating machinery such as gearboxes, especially when they are operating under varying load conditions. Varying load conditions amplitude modulate the measured vibration signal

and cause the rotation speed of the system to change. The change in system speed results in frequency modulation of the gear mesh frequency (Stander et al., 2001; Stander et al., 2002[a]; Stander et al., 2002[b]).

When using vibration signatures for condition monitoring of gearboxes it is difficult to extract meaningful information from raw time domain vibration data. This is because the characteristic frequencies generated by newly developed faults in gearboxes can be very low in amplitude and are therefore often overshadowed, or masked, by other vibration components such as random noise and interference from additional vibration sources in the machine or neighbouring machines. To overcome this problem the vibration signal is sampled at a frequency that is synchronised exactly with the rotation of the gear of interest and the samples obtained for each singular position of the gear are then ensemble-averaged. When sufficient averages are taken, all the vibration from the gearbox, which is asynchronous with the vibration of the gear, cancels out, leaving only the vibration produced during one rotation of the gear of interest. Local variations in the meshing pattern and modulation in the gear of interest are therefore made visible (McFadden, 1987; McFadden et al., 1985; Stewart, 1977). This procedure is called time domain averaging or synchronous averaging.

The resulting time synchronously averaged signal obtained through the time domain averaging process indicates the vibration produced during one rotation of the monitored gear. The synchronous vibration signal can be related to the meshing stiffness of the gear being monitored. Variations in the meshing stiffness of the gear indicate wear and or incipient local defects that are related to a variation in gear teeth stiffness. Time domain averaging is an extremely effective technique, but it requires an enormous amount of vibration data to calculate. This problem makes time domain averaging less attractive on online gearbox condition monitoring system. The challenge remains to develop a synchronous time domain averaging filter that reduces the amount of vibration data that is required for direct synchronous time domain averaging of gear vibration data. A reduction in the amount of input gear vibration data required for synchronous time domain averaging of gear vibration brings us closer to the successful implementation of synchronous time domain averaging on an online gearbox condition monitoring system.

1.2 Literature survey

The objective of this research is to develop a filter for synchronous time domain averaging of gear vibration using computational intelligence. The purpose of the filter is to reduce the amount of gear vibration data that needs to be stored in the data acquisition system in order to calculate the time domain average of a gear vibration signal. The literature survey addresses the following topics:

- Signal processing techniques for early detection of gear failure through vibration measurements.
- Digital filtering.
- Application Artificial Neural Networks and Support Vector Machines in pattern recognition.

1.2.1 Signal processing techniques for early detection of gear failure through vibration measurements.

In this section, as background, different signal processing techniques for early detection of gear failure through vibration measurements are discussed. The underlying premise of vibration analysis is that changes in the mechanical condition of the system produce changes in the vibration that the system produces. In extremely simple systems, these changes take the form of an increase in the amplitude of the total vibration, which can be easily detected with simple instruments. For more complex systems, changes in the total vibration due to the deterioration of a single machine element are less significant and more sophisticated vibration-processing techniques are needed to detect the damage (McFadden, 1987).

One of the most popular techniques for early detection of gear failure through vibration measurements over the last four decades has been spectral analysis, in which the amplitude spectrum of the measured vibration spectrum is measured and displayed. Spectral analysis is a particularly powerful technique because different elements of a mechanical system generally produce vibration at different frequencies.

In 1977 Stewart presented some useful data analysis techniques for gear diagnostics. These techniques enhance the clarity of the changes on the time domain average using digital signal processing, by removing the normal vibration from the time domain

average. In one of these techniques all of the tooth meshing components and their harmonics are eliminated from the spectrum of the time domain averaging and the remaining time signal is reconstructed to produce the "residual" signal. Stewart showed that the residual signal often shows evidence of a defect long before it can be seen in the time domain average.

In 1985 McFadden and Smith applied modulation theory to a model of gear vibration and showed that band pass filtering the "residual" signal about the dominant meshing harmonics and developing the envelope produced a function that describes the amplitude and phase modulation present in the original averaged signal. The application of this technique to the vibration produced by a gear known to contain incipient fatigue cracks suggested that this method is highly effective, and demonstrate that the phase modulation of the vibration is a more important indicator of a crack than amplitude modulation.

McFadden (1986) illustrated that the signal average can be completely demodulated by simple signal processing techniques to produce separate approximations to the amplitude and phase modulation functions. He demonstrated the effects of both early and advanced fatigue cracks on the modulation functions using signal averages of vibration on spiral bevel pinion in a helicopter gearbox.

In another publication McFadden (1987) presented time domain averaging as an alternative approach for early detection of failure in gears. This author stated that, if a second signal is acquired which is synchronised with the rotation of the gear of interest, and the ensemble average of the vibration is calculated with the start of each frame being determined by the synchronised signal, all the vibration that is asynchronous with the rotation of the gear cancels out, leaving an estimation of the vibration of the gear of interest during one gear revolution. Time domain averaging therefore reduces a complex system such as a gearbox into a simpler system as it eliminates vibration from other system element (McFadden, 1987).

In his paper White (1991) demonstrated the use of a signal demodulator unit for extracting meaningful information from rolling element bearing and gearbox vibration for predictive maintenance.

Wang and McFadden (1993) examined the application of the spectrogram to calculate the time-frequency distribution of gear vibration signals. The spectrogram represents the energy distribution of the signal over the frequency and time. Their results suggest that the spectrogram may provide a powerful tool for the early detection of local gear damage.

In a later publication Wang and McFadden (1995) investigated the use of the orthogonal wavelet transform to detect the abnormal transients generated by early gear damage from gearbox casing vibration. Orthogonal wavelets, such as Daubechies 4 and harmonic wavelets, are used to transform the time domain synchronous vibration signal into the time-scale domain. The orthogonal wavelet transform uses fast algorithms and decomposes the signal into the minimum number of wavelets series. These authors discovered through comparison with non-orthogonal wavelet transform for same length of discrete data, that the description of the signal in the 3-dimensional map of the wavelet transform is not sufficiently comprehensive due to limited scales.

McFadden et al. (1999), described the generalised S transform, a variant of the wavelet transform, which allows the calculation of the instantaneous phase signal, and its application to decomposition of vibration signals from gearbox systems for early detection of failure. They demonstrated the decomposition of a signal using the generalised S transforms and a new window function with a numerically generated test signal and experimentally measured gear vibration data.

Baydar and Ball (2000) used another time-frequency distribution called the Instantaneous Power Spectrum (IPS) in the detection of local faults in helical gears. Their paper describes the IPS and then examines its capability of extracting condition indicating information from gear vibration signals and also assessing the severity of the fault. The paper further examines the ability of the IPS to detect faults under varying load conditions. Their results show that the IPS can be used to detect faults both under constant and varying load conditions.

In a later study Baydar and Ball (2001) conducted a comparative study of acoustic and vibration signals in the detection of gear failure using Wigner-Ville distributions. Their

results suggest that acoustic signals are very effective for early detection of faults and may provide a powerful tool for indicating the various types of progressing faults in gears.

Staszewski and Tomlinson (1994) presented an application of the wavelet transform to fault detection in spur gears. In further work these authors use a moving window procedure for local fault detection in gearboxes (Staszewski and Tomlinson, 1997).

Staszewski et al. (1997) presented a study of the use of the Wigner-Ville distribution in gearbox condition monitoring. In contrast to other applications of the Wigner-Ville distribution, their paper reported on the application of statistical and neural network pattern recognition procedures to reliably detect gear tooth faults.

Wang and Wong (2000) developed a linear prediction method that is based on the assumption that the vibration caused by a sound pair of gears can be modelled as a stationary autoregressive process. These authors stated that the approach is independent of the operating conditions, but the precise influence of varying loads is not documented. The results of their paper indicate that the linear prediction method can be used effectively in the detection and diagnosis of gear failure.

Stander and Heyns (2001) noted the influence of varying loads on vibration monitoring of gears. Stander et al. (2002)[b] conducted an experimental investigation to observe the influence of fluctuating load conditions on the measured acceleration signal. They concluded that the load variation manifests itself as a low-frequency modulation on the measured acceleration signal. In another publication Stander and Heyns (2002)[a] investigated the use of the Instantaneous Shaft Speed (ISS) in condition monitoring of gearboxes. They postulated that the integrity of the gear tooth in the mesh could be monitored through the utilisation of the ISS measurement. The authors concluded that a natural separation between different levels of damage could be obtained by monitoring the instantaneous gear shaft speed under various fluctuating load conditions.

Paya et al. (1997) investigated the use of artificial neural networks based fault diagnostics of rotating machinery using wavelet transforms as a pre-processor. The real time domain vibration signal obtained from the gearbox transmission were pre-

processed by wavelet transforms for neural networks to perform fault detection and identify the exact kind of fault occurring in the transmission. They showed that by using multi-layer artificial neural networks on the set of data pre-processed by wavelet transforms, single and multiple faults could be successfully detected and classified into specific groups.

Zacksenhouse et al. (2000) conducted a series of tests on a helicopter transmission for the purpose of generating a database that can be used to evaluate general diagnostic tools, particularly neural networks. They demonstrated that the meshing vibrations induced by a large collector gear located on the quill shaft are significant and may interact with the vibrations induced by other elements attached to the same shaft. An appropriate model is developed and the effect of the collector gear, called cross-gear-pair interaction, is studied using different signal processing tools.

Decker (2002)[a] conducted a survey of standard vibration diagnostic parameters for crack detection in spur gears used in the Health and Usage Monitoring Systems (HUMS). The results of his study indicated that detection methods used in HUMS are not robust or repeatable. The cracks actually progressed at a much faster rate than anticipated reducing the available time for detection.

In another study Decker (2002)[b] proposed a new gear failure analysis feature and two new detection techniques. The time synchronous averaging concept was extended from being revolution-based to tooth-engagement based. The detection techniques were based on statistical comparison among the averages for the individual teeth. The results indicated that these techniques do not produce an indication of damage that significantly exceeds experimental scatter.

Dempsey et al. (2002) developed a diagnostic tool for detecting damage on a spiral bevel gear by integrating two different monitoring technologies, oil debris analysis and vibration analysis. Their results showed that combining vibration and oil debris measurement technologies improves the detection of pitting damage on spiral bevel gears.

The literature indicates that gear condition monitoring is now in its mature stages. Almost all the vibration analysis methods mentioned above require some form of pre-processing with synchronous time domain averaging to increase their diagnostic capabilities. It is because of this very reason that efficient methods for synchronous time domain averaging are required.

1.2.2 Digital filtering

Signal processing can be defined as the processing performed on signals to extract useful information. Of the many signal processing methods currently available, digital filtering is one of the most powerful. In order to understand the principles used in the development of the synchronous filter for time domain averaging it is necessary to understand of the fundamental theory and some of the recent developments of digital filter technology. In this section a brief history and some relevent applications of filter technology are presented.

Digital filters evolved from simulation of analog filters on the early digital computers of the 1940s. Their first application was in geological exploration of oil fields where the data was collected and stored for future processing. The seismologists found that analog signal-processing methods did not help them distinguish signal from noise. However, through discrete convolution and other noise elimination techniques, they were able to process the seismograms digitally to yield a filtered form that was much easier to interpret, and thus new oil sources were identified.

In signal processing, the function of a filter is to remove unwanted parts of the signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range. This basic idea can be illustrated by the block diagram in Figure 1.

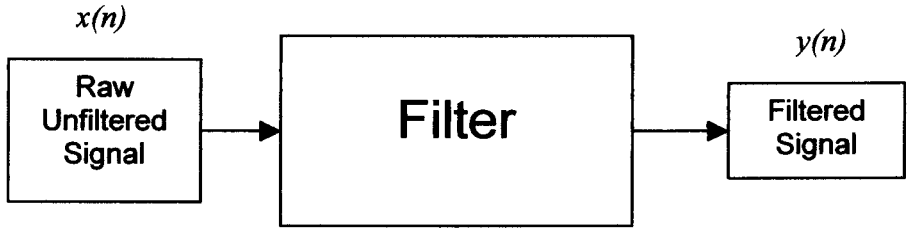


Figure 1 Basic filter concept

In general a filter takes an input sequence *x(n)* and produces an output sequence *y(n)* as shown in Figure.1. More recent developments in filter technology include the development of intelligent filters and adaptive filters. Some relevant filter developments are presented in the following paragraphs.

Zhong et al. (2003) tackled the fault detection problem in dynamic systems with modelling errors and unknown inputs. In this paper the robust fault detection filter design problem for uncertain linear time-invariant systems with both unknown inputs and modelling errors is studied. The main results include the development of an optimal reference residual model, the formulation of robust fault detection filter design problem, the derivation of a sufficient condition for the existence of a robust fault detection filter and its construction based on the linear matrix inequality solution parameters, and the determination of adaptive threshold for fault detection.

Augustyn et al. (2003) presented a new method for filtering signals using the Modified Recursive Discrete Fourier Transform (MRDFT). The basic idea of this method is the application of the user-defined context to the recursive form of the Discrete Fourier Transform (DFT) and filtration data or signals. The context is defined in the frequency domain and the mathematical implementation of the context in a recursive DFT is presented. The method is controlled by an intelligent decision making system, which decides what context present in predefined base of context can be applied to the algorithm. This means that the filtration process extracts only the desired signal features.

Another important concept in the context of this work is that of averaging. Braun (1975) analysed the extraction of a periodically repeating signal from noise coherent averaging. He considered the averaging process as a filtering process and conducted most of his analysis in the frequency domain. He described a general approach for dealing with digital comb filters, enabling the design and analysis of related signal processors.

McFadden (1987) showed that the comb filter model for time domain averaging does not correctly describe the extraction of periodic waveforms from additive noise because

it assumes knowledge of the signal over an infinite time and the result it produces is not exactly periodic. He presented a revised model which requires only a finite number of samples of the signal and which produces a result that is periodic. He also demonstrated that the rejection of periodic noise of a known frequency could be optimised by the appropriate selection of the number of averages.

Moczulski (1987) described the digital synchronous filtering technique. The digital synchronous filtering technique makes it possible to estimate the time history of periodic components of the signal being analysed and the corresponding frequencies which are integer multiples of some triggering frequency. The signal components, which are not synchronous with the triggering signal, are simultaneously attenuated. The digital synchronous filtering technique is based upon the time domain averaging technique. The bank of filters obtained by Moczulski makes it possible to estimate the averaged time courses of the periodic components of the signal and the amplitude and phase characteristics of the filters are given. Only simple fixed-point arithmetic operations were used in order to prepare the necessary software for signal processing.

McFadden (1989) presented an interpolation technique for time domain averaging of gear vibration by digital computer. This technique provides an alternative to the phase-locked frequency multiplier for the calculation of the time domain average of gear vibration signals. Higher-order interpolation techniques produce flatter pass bands and lower side lobes in the stop band but require longer calculation times. Aliasing errors are introduced into the result by replication of the side lobes during interpolation, but in general are attenuated by time domain averaging.

1.2.3 Application of artificial neural networks and support vector machines in pattern recognition

Artificial Neural Networks (ANNs) can be defined as an information-processing paradigm inspired by the way the densely interconnected, parallel structure of the human brain processes information. They are also referred to by other names, such as connectionism, parallel distributed processing, neuro-computing, natural intelligent systems and machine learning algorithms. The key element of the ANN paradigm is the novel structure of the information processing system. It is composed of a large number

of highly interconnected processing elements that are analogous to neurons and are tied together with weighted connections that are analogous to synapses.

There are numerous neural networks that have been investigated. Some of the more popular neural networks include the Perceptron, Multi-Layer Perceptron (MLP), Learning Vector Quantization (LVQ), Radial Basis Function (RBF), Hopfield networks, and Kohonen's self-organizing feature maps (SOM). ANNs are also classified as feedforward, or recurrent (implement feedback) depending on how data are processed through the network. Sometimes ANNs are classified by the method of learning or training they use. Some ANNs such as MLP and RBF employ supervised training, in which the network's error function minimisation involves both the input and the target values. Other ANNs such as the SOM networks employ unsupervised learning, which only involves the input during the training. ANNs are attractive in digital signal processing for the following reasons:

- ANNs can form arbitrary decisions so that any complex mapping from a set of noise-contaminated signal to a noise free signal can be realized.
- ANNs can easily be implemented as software or in specialized hardware.
- ANNs are quite resilient against distortions in the input data and have a capability to learn and generalize when properly trained.
- ANNs are often good at solving problems that are too complex for conventional technologies and are often well suited to problems that people are good at solving, but for which traditional methods are not suitable, such as character recognition.

Neural networks have found extensive application in pattern recognition, signal classification, and image processing. In this work the idea is to apply ANNs and SVMs in a pattern recognition or predictive task. The idea is to train the ANN to predict the ensemble average (time domain average) of a large input matrix (rotation synchronised gear vibration signals) without using the entire input matrix. The successful application of ANNs and SVMs in this sense holds potential of massive reduction in the amount of data required in the synchronous time domain averaging task. In this application, this means a reduction in the amount of data that needs to be collected and stored in the data

acquisition systems before the synchronous time domain averaging can be calculated. Some of the more popular ANN formulations for applications related to this work include the MLP, RBF networks and more recently the Support Vector Machines (SVMs). Some of the most relevant applications to this work include work by the following researchers:

Gaudart et al. (2002) compared the performance of MLP and linear regression (LR) with regards to the quality of prediction and estimation and the robustness to deviations from underlying assumptions of normality, equality of variance and independence of errors. The comparison between connectionist and linear models was achieved by graphic means including prediction intervals, as well as by classical criteria including goodness-of-fit and relative errors. The empirical distribution of estimations and the stability of MLP and LR were studied by re-sampling methods. MLP and LR comparable performance and robustness despite the flexibility of the connectionist models.

Gardner and Dorling (1999) trained MLP neural networks to model hourly NO_x and NO_2 pollutant concentrations in Central London from basic hourly meteorological data. Their results show that the models perform well when compared to previous attempts to model the same pollutants using regression based models. Their work also illustrates that MLP neural networks are capable of resolving complex patterns of source emissions without any explicit external guidance.

Walde et al. (2003) investigated the impact of sample size and sample randomness on the predictive accuracy of MLP. The MLP proved to be useful for classification problems although they are dependent on the sample size and the non-linearity of the underlying problem. A saturation curve describes the dependency of the network performance on the sample size used. This function enables the user to evaluate the achieved network performance and the usefulness of additional data. It is demonstrated that the network leads to narrower confidence intervals of the performance measures in comparison to classical methods even for small sample sizes. The experimental results show the validity of the law, for even relatively small sample sizes, that the standard error of the hit ratio decreases by one over the square root of the sample size.

Taurino et al. (2003) showed the capability of a sol-gel based electronic nose to be used in qualitative and quantitative analysis with the aim to recognize common volatile compounds usually present in the headspace of foods. They showed how linear technique, such as the Principal Component Analysis (PCA) algorithm can be used for inspecting data distribution in simple cases like cluster discrimination. They also used MLP and RBF networks for difficult non-linear regression problems. Their results showed that the MLP gives better performance for their application.

Fidêncio et al. (2003) used the RBF and MLP networks for non-parametric regression of organic matter content in soils determined by conventional chemical measurements and by diffuse reflectance spectra in the near infrared region. The observed results using RBF were better than those obtained by Partial Least Squares (PLS) regression and MLP feedforward networks with a back-propagation learning algorithm. These authors concluded that RBF is a suitable tool for their application, with additional advantages over MLP, since the training procedure is less dependent on the initial conditions.

Alsing et al. (2002) introduced a multinomial selection problem procedure as an alternative to classification accuracy and receiver operating characteristic analysis for evaluating competing pattern recognition algorithms. The multinomial selection problem procedure demonstrates increased differentiation power over traditional classifier evaluation methods when applied to three "toy" problems of varying difficulty. The multinomial selection problem procedure is also used to compare the performance of statistical classifiers and artificial neural networks on three real-world classification problems. The results provide confidence in the multinomial selection problem procedure as a useful tool for distinguishing between competing classifiers and providing insights on the strength of conviction of a classifier.

Another promising method for tackling regression and prediction problems is Support Vector Machines (SVMs). Yang et al. (2002) applied the SVMs in financial prediction of noisy, time-varying financial data. Their experimental results showed that the use of standard deviation to calculate a variable margin results in a good predictive result in the prediction of Hang Seng Index.

Ramesh et al. (2003) presented a hybrid Support Vector Machines–Bayesian Network (SVM-BN) model that seeks to address the issue that most error models developed thus far generally employ neural networks to map the input to the output and give no account for the specific conditions that apply to the process being modelled. In their model the experimental data is first classified using a Bayesian Network model. Once the classification has been effected, the error is predicted using a SVM model. Their hybrid error model thus predicts the error according to the specific operating conditions. This concept leads to a more generalised prediction model.

Gunn (1998) wrote a comprehensive technical report on the support vector machines for classification and regression. Another similar publication is a tutorial by Burges (1998) entitled Tutorial on Support Vector Machines For Pattern Recognition in which the author presents in the light of regression and classification problems.

1.3 Research objectives

The literature indicates that pre-processing gear vibration data with synchronous time domain averaging can increase the diagnostic capabilities of the measured gear vibration data. This is important to the engineer because it bears the potential of increasing the reliability, repeatability and the diagnostic capability of gearbox condition monitoring strategies. Time Domain Averaging (TDA) is an extremely effective technique for the extraction of periodic data from the vibration signals of rotating machinery. TDA, however, requires an enormous amount of gear vibration data to calculate. This makes it unattractive for on-line gearbox condition monitoring systems.

The literature also indicates that ANNs and SVMs can be successfully used in the non-linear mapping of some input space to an output space. This observation is important to this work in that it bears the potential of reducing the amount of input gear vibration that is required for calculating the TDA of the gear vibration. The time domain averaging process can itself be viewed as a broadband noise filter that eliminates all the vibration that is asynchronous to the vibration of the gear of interest.

The purpose of this study is therefore to investigate and develop a synchronous filter for time domain averaging of gear vibration data using of ANNs and SVMs.

- The developed filter should provide a considerable reduction in the amount of vibration data that needs to be collected and stored in the data acquisition system in order to calculate synchronous time domain average of a gear vibration signal.
- The developed synchronous time domain averaging filter model should retain the diagnostic enhancement capabilities of the TDA calculated by direct averaging.
- The filtering and diagnostic capabilities of the developed filter should be validated for both constant and varying load conditions on experimental gear vibration data.
- Comparison should be made between the performance of the developed filter model and direct time domain averaging.

1.4 Document overview

The theory and mathematics of existing time domain averaging models is presented in Chapter 2. This chapter also presents some simulations that highlight the strengths and weaknesses of some of the popular time domain averaging models.

In Chapter 3 the theory of the MLP neural network, the RBF neural network and SVMs in the context of this work is presented. Simulations on experimental gear vibration data are conducted to investigate the suitability of these formulations for application in the synchronous time domain averaging filter model.

Chapter 4 presents the development process of the synchronous filter for time domain averaging of gear vibration data. Two different synchronous filtering models are developed. Gear vibration data from previous tests is used to investigate the influence of different model parameters on the prediction capability for each of the developed synchronous filtering models.

In Chapter 5 the developed synchronous filtering models are tested on experimental data from accelerated gear life test rig for constant and varying load conditions. The results confirm the suitability of the developed synchronous filtering models for time domain averaging of gear vibration. A comparative study of these models is presented

In Chapter 6 the conclusion to the research and recommendations for further work are presented

Chapter 2

Time domain averaging models

2.1 Introduction

Time domain averaging is a signal processing technique that may be used to extract the synchronous periodic content of a measured vibration signal from the measured vibration signal. This process requires either accurate knowledge of the repetition frequency of the desired frequency, if periodic, or else a second signal that is synchronous with the first signal, but free of noise. Using either the repetition frequency or the synchronous signal, successive blocks of the noisy signal may be sampled and ensemble averaged. When sufficient averages are taken, it is found that the noise in the ensemble averaged signal cancels out, leaving an improved estimate of the desired repetitive signal (McFadden, 1989). One important application in which the periodic signal must be extracted from the noise is the mechanical engineering problem of analysis of vibration from gearboxes. When analysing gearbox vibration, it is sometimes necessary to extract a periodic signal such as the tooth meshing vibration of a single gear from the vibration of the machine. Some understanding of the time domain averaging technique is required for the analyst to appreciate its limitations and successfully optimise its performance for a particular application. This chapter, therefore, presents some of the most commonly used time domain averaging models.

2.2 Existing models

2.2.1 Comb filter model

For many years, time domain averaging has been modelled by the convolution of the noisy signal with a finite train of impulses in which the time between the impulses is equal to the period of the desired signal. It has been shown that this process is equivalent in the frequency domain to the multiplication of the Fourier transform of the noisy signal by a comb filter, thus passing only the frequency components which fall at the fundamental and harmonic frequencies of the desired signal (Trimble, 1968). In this

section the application of the comb filter model for time domain averaging in the extraction of periodic signals is presented.

It can be shown that calculation of the synchronous time domain using a trigger signal having a frequency f_t is equivalent to the convolution

$$y(t) = c(t) * x(t) \tag{2.1}$$

where $c(t)$ is a train of N impulses of amplitude $1/N$, spaced at $T_t = 1/f_t$, given by

$$c(t) = \frac{1}{N}\sum_{n=0}^{N-1}\delta(t + nT_t). \tag{2.2}$$

The convolution of *c(t)* and *x(t)* is given by

$$\begin{aligned} c(t) * x(t) &= 1/N\sum_{n=0}^{N-1} x(t + nT_t) \\ &= [x(t) + x(t + T_t) + \mathrm{L} \ + x(t + (N-1)T_t)]/N. \end{aligned} \tag{2.3}$$

The time domain average *y(t)* of the signal is then defined by

$$y(t) = \frac{1}{N}\sum_{n=0}^{N-1} x(t + nT_t). \tag{2.4}$$

This equation has the same form as that of the existing comb filter model (Braun, 1975; McFadden, 1987). In the frequency domain this is equivalent to the multiplication of the Fourier transform of the signal $X(f)$ by the Fourier transform of the impulse signal $C(f)$. This operation is represented by

$$Y(f) = C(f) \cdot X(f). \tag{2.5}$$

The Fourier transform of $c(t)$ is $C(f)$, which is a comb filter function of the form

$$C(f)=\frac{1}{N}\frac{\sin(\pi NT_t f)}{\sin(\pi T_t f)}. \tag{2.6}$$

Increasing the value of N in Equation (2.6) narrows the teeth of the comb. For very large values of N, only frequencies at exact multiples of the trigger frequency f_t are passed. Equation (2.6), therefore, implies that in the frequency domain, for large values of N, synchronous time domain averaging can be viewed as a complete removal of all components within the signal that occur at integer multiples of the trigger frequency f_t.

Figures 2.1 (a) to 2.1 (d) show a form of the amplitude spectrum $|C(f)|$ for $N=1,2,4$ and 8 of a comb filter plotted against the normalised Nyquist frequency. The spectrum takes the form of a comb with the teeth of the comb spaced at intervals $f_t=1/T_t$. The teeth of the comb have unit amplitude regardless of N.

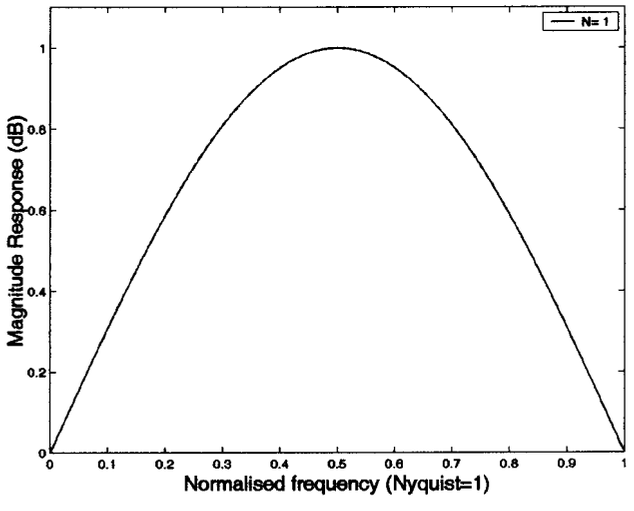


Figure 2.1 (a) Amplitude response of comb filter vs. normalised Nyquist frequency for $N=1$.

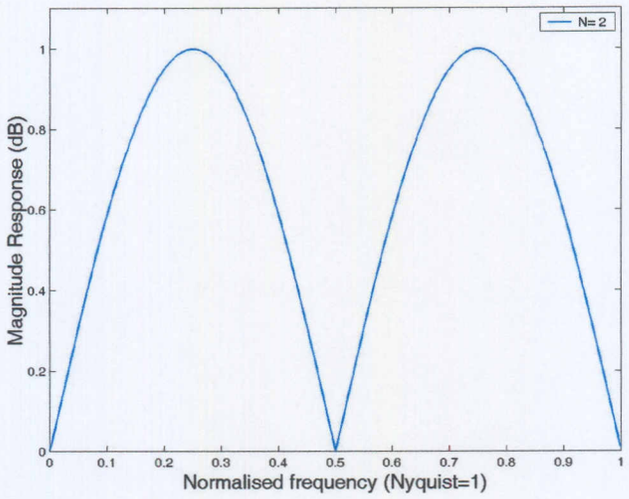


Figure 2.1 (b) Amplitude response of comb filter vs. normalised Nyquist frequency for $N = 2$.

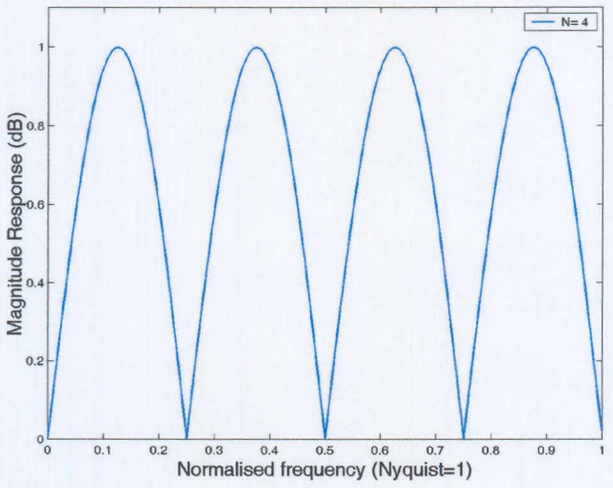


Figure 2.1 (c) Amplitude response of comb filter vs. normalised Nyquist frequency for $N = 4$.

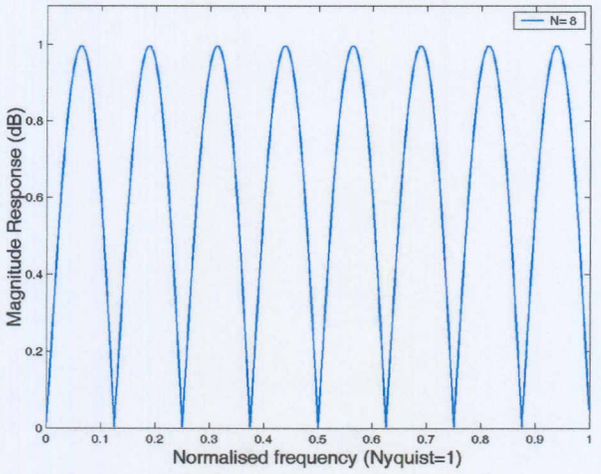


Figure 2.1 (d) Amplitude response of comb filter vs. normalised Nyquist frequency for $N = 8$.

There are two features of the comb filter model that restrict its application to the extraction of periodic waveforms using digital computers. The first of these factors is that in the comb filter model there are bounds placed on the time signal. The model assumes that the signal *x(t)* is known over infinite time *t* and that the time domain average *y(t)* is defined over all time *t*, even though only a finite number of averages are calculated. In practice, the signal *x(t)* can only be defined over a finite time. Noise components that are not harmonically related to the repetition frequency f_t may be passed by the comb filter, therefore, the estimate of the time domain average will not be exactly periodic.

Figure 2.2 shows the performance of a comb filter in the extraction of a signal and its harmonics from a numerically generated signal $z(t) = x_p(t) + e(t)$ where $x_p(t)$ is known periodic component defined by:

$$\begin{aligned} x_p(t) = \; & \sin(2\pi 50t) + \sin(2\pi 100t) - 0.45\cos(2\pi 200t) + 2.1\sin(2\pi 150t) + \ldots \\ & \sin(2\pi 600t) - 2.5\cos(2\pi 250t) + \sin(2\pi 300t) + \sin(2\pi 350t) + \ \ldots \\ & \cos(2\pi 400t) + \sin(2\pi 450t) + \sin(2\pi 500t) + 0.25\cos(2\pi 550t) \end{aligned} \tag{2.7}$$

where *t* = (1: 0.001: 6), and the additive noise component *e(t)*, the noise content of the signal is defined by:

$$\begin{aligned} e(t) = \; & 0.3randn(6000) - 0.48randn(6000) + 0.1randn(6000) + \ldots \\ & 0.33randn(6000) + 0.17randn(6000), \end{aligned} \tag{2.8}$$

where *randn* is a MATLAB function that defines a set of normally distributed random numbers selected from a normal distribution with a mean of zero and variance of one.

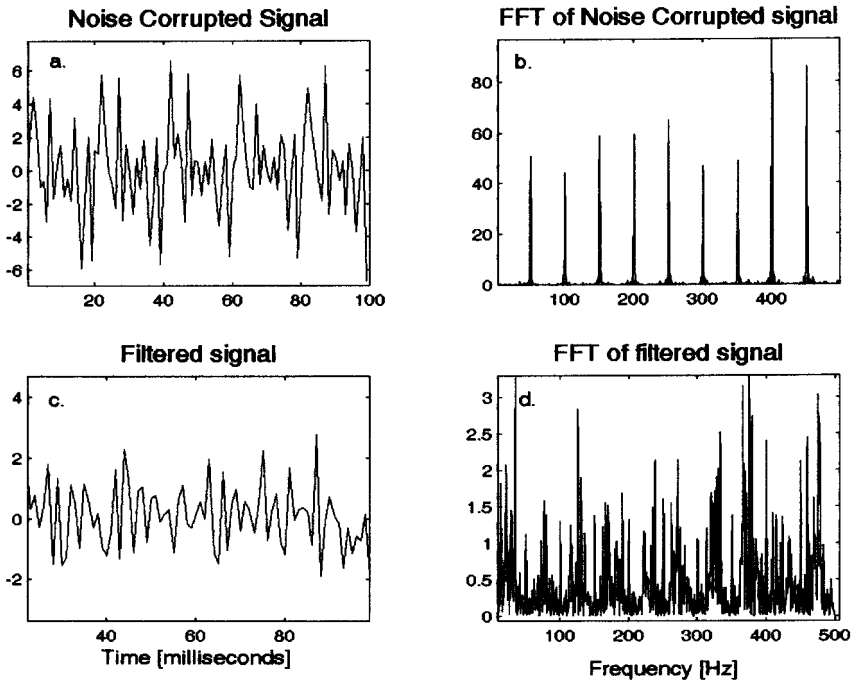


Figure 2.2 (a) Noise corrupted signal; (b) FFT of noise corrupted signal; (c) Signal filtered with a comb filter to remove 50 Hz and all the related harmonics; (d) FFT of filtered signal.

From Figure 2.2 (c) it is observed that the amplitude of the signal decreased after filtering but nothing much can be said about the frequency content of the resultant signal. Figure 2.2 (d) shows the frequency spectrum of filtered signal. It is observed that the amplitude of the frequency spectrum has decreased from 80 to 3 and 50 Hz and all its harmonics have been filtered out leaving only the noise content that does not coincide with 50 Hz.

2.2.2 Double comb filtering

Another attractive model for extracting the time domain averaging is the double comb filtering approach as documented by Braun and Seth (1980). In this model the time domain signal *x(t)* is decomposed as follows:

$$x(t) = x_p(t) + x_n(t) + n(t) \tag{2.9}$$

where $y_p(t)$ is the periodic term and $x_p(t) = x_p(t+T)$; $x_n(t)$ is a random repetitive term and $n(t)$ is a residual term that is time-locked to a basic period T. A truly periodic

component can only be generated by a truly periodic mechanism. This can only exist in an ideal system because any process involving some sort of friction, liquid or gas flow, and some non-reversible fatigue processes would include to some extent a component like $x_n(t)$ (Braun and Seth, 1980). For a "more or less" periodic process this would show gross periodic character, but no exact repetition of the nature $x_p(t) = x_p(t+T)$ would occur. The parameter $x_n(t)$ thus describes a repetitive non-periodic process as opposed to $x_p(t)$. The periodic component $x_p(t)$ can be computed using the comb filter model as described in Section 2.2.1. The extraction of the random repetitive term $x_n(t)$ is based on the expression

$$y_n(t) = g(t)x_r(t). \tag{2.10}$$

In Equation (2.10) x_r denotes a continuous random process of no obvious time pattern, and $g(t)$ is a deterministic periodic function of period T, (i.e. $g(t) = g(t + T)$). For a case where $x_n(t)$ is derived from a narrow band continuous process, where narrow band refers to a band limited process whose bandwidth is small relative to its centre frequency, where $x_r(t)$ contains negligible energy above frequency f_{max} such that

$$2f_{max} \leq f_T = 1/T \tag{2.11}$$

a possible computation scheme for detecting the components of $x_n(t)$ consists of using a comb filter tuned to the narrow bands located around multiples of the fundamental frequencies. A schematic diagram of the computation scheme is shown in Figure 2.3.

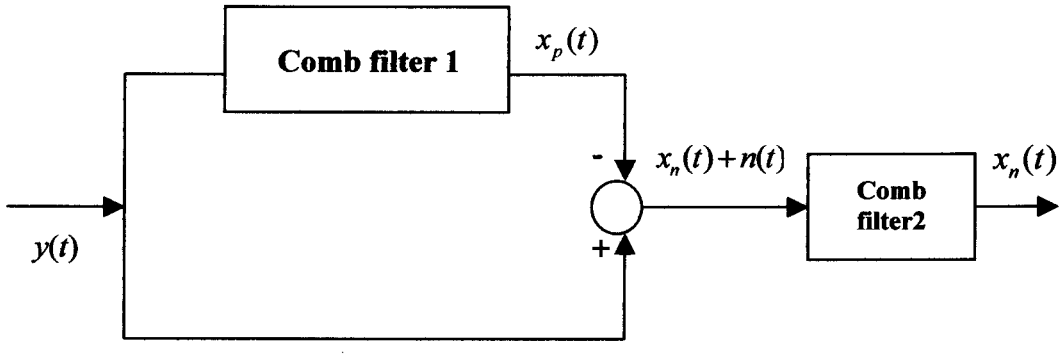


Figure 2.3 Double comb filtering model

The first stage is used to compute $x_p(t)$ and then $x_p(t)$ is subtracted from the original signal $x(t)$. After subtraction, a second comb filter extracts $x_n(t)$. Both these stages will be computed for the same period T.

2.2.3 Revised window model

The revised window model was suggested by McFadden (1987) to address some of the problems encountered with the comb filter described in Section 2.2.1. This model overcomes the problems with the comb filter model in that it requires knowledge of only a finite block of the noisy data and it produces a result that is exactly periodic. This model includes the effect of the signal's sampling frequency f_s. In this section the revised window model is briefly discussed.

Consider a rectangular window $\upsilon(t)$ of unit amplitude and width T_R centred at $t=0$ with Fourier transform $V(f)$ are defined by

$$V(f)=T_R \sin(\pi T_R f)/(\pi T_R f). \tag{2.12}$$

Shifting the window $\upsilon(t)$ to the positive direction by an amount $(T_R/2)-(T_s/2)=(T_R-T_S)/2$ where $T_S=1/f_S$ is the period between the samples of the input signal therefore the edges of the window are located midway between the sampling of the impulses. This avoids the problem of an impulse occurring at the edge of the window. The shifted window $w(t)$ and its Fourier transform $W(f)$ are, respectively, defined by

$$w(t)=\upsilon(t-(T_R-T_S)/2) \tag{2.13}$$

$$W(f)=V(f)e^{-i\pi f(T_R-T_S)}. \tag{2.14}$$

Consider now the sampling of the signal $x(t)$ at a frequency f_S over the window of duration T_R. The window, defined by $w(t-nT_R)$, consists of the window $w(t)$ shifted by $t=nT_R$. The sampling of the signal is produced by multiplication of $x(t)$, $w(t)$ and $c(t)$, where $c(t)$ is the pulse signal defined by Equation (2.2). The result is convolved

with the unit impulse by $\delta(t+nT_R)$ located at $t=-nT_R$, thus performing a shift of the sampled signal by nT_R in the negative time direction. The result is given by

$$\delta(t+nT_R)*[x(t)\cdot w(t-nT_R)\cdot c(t)]=x(t+nT_R)\cdot w(t)\cdot c(t+nT_R). \tag{2.15}$$

Now the function $c(t)$ is periodic in T_S. If T_S is chosen such that an integral number M of samples is taken per repetition period T_R then $T_R=MT_S$. This implies that $c(t)$ will also be periodic to T_R, so that $c(t)=c(t+nT_R)$. By replacing $c(t+nT_R)$ in equation (2.15) one gets

$$\delta(t+nT_R)*[x(t)\cdot w(t-nT_R)\cdot c(t)]=x(t+nT_R)\cdot w(t)\cdot c(t). \tag{2.16}$$

An estimate of time domain average $a(t)$ is given by

$$a(t)=1/N\sum_{n=0}^{N-1}x(t+nT_R)\cdot c(t)\cdot w(t), \tag{2.17}$$

which is equivalent to

$$a(t)=c(t)\cdot w(t)/N\sum_{n=0}^{N-1}x(t+nT_R), \tag{2.18}$$

therefore, the revised window model is defined by

$$a(t)=s(t)\cdot w(t)\cdot x(t). \tag{2.19}$$

Note that although $x(t)$ is not bound in time, $a(t)$ is bounded in time because of the effect of the window $w(t)$. This model therefore satisfies the requirement of knowledge of the signal over only a finite time. Analysis of $a(t)$ in the frequency domain by convolution theorem (Bringham, 1974) shows that $a(t)$ can be forced to be periodic by sampling its Fourier transform $A(f)$ in the frequency domain. This is achieved by multiplying $A(f)$ by an infinite train of ideal impulses $R(f)$, with the impulses spaced

at a repetition frequency f_R. An estimate of the revised window time domain averaging model is given by

$$h(t) = a(t) * r(t) \tag{2.20}$$

The revised window model remarkably changes the result that is predicted by the original comb filter model. Over and above requiring knowledge of the signal over finite time, it also ensures that the obtained result is periodic.

2.2.4 Using direct averaging

Again consider a signal $z(t)$ composed of a periodic signal $x_p(t)$ with known period T_R and an additive noise component $e(t)$

$$z(t) = x_p(t) + e(t). \tag{2.21}$$

The periodic component $x_p(t)$ of signal $z(t)$ can be extracted by direct time domain averaging. To calculate the direct time domain average of a vibration signal, a rotational signal from a sensor mounted on the input shaft or some other suitable location on the rotating machine is used. This rotational signal is used either to control the sampling of the total vibration signal or to determine the accurate period of the vibration of the component of interest and to separate out that vibration component. When the rotational signal is used to control the sampling of the total vibration, a phase locked frequency multiplier is used to convert the rotational signal to the required sampling control signal, which consists of a pulse train synchronised with the rotation of the required gear. When the rotational signal is used to determine the accurate period of the vibration of the component of interest, both the total vibration signal and the rotational signal are sampled simultaneously at fixed clock frequency. When monitoring a gearbox, the rotational signal can be obtained from a sensor like a shaft encoder mounted on the input shaft to the gearbox. The accurate period of the rotating signal can be easily obtained from the shaft encoder signal, and the accurate period of the vibration of the required gear can then be calculated using the transmission ratio.

After the correct period of the rotating signal has been obtained, there are two different approaches that can be followed to calculate the time-domain average of the vibration to separate out the required vibration component. The first approach is by directly averaging some segments of the total vibration (Braun, 1975, Braun and Seth, 1980). The second approach is to first interpolate the total vibration and resample it in the interval that can exactly divide the calculated period, and then averaging the interpolated signal (McFadden, 1989).

In direct time domain averaging, the rotational signal obtained from a sensor mounted on the input shaft to the component of interest is used to synchronise the measured vibration signal with the rotation of that component. This operation gives the vibration produced by that specific rotating component over each rotation. The vibration signals from the rotations are simply averaged to obtain the time domain average after k revolutions. Figure 2.4 illustrates the direct time domain averaging procedure using vibration data from the accelerated gear life test rig developed by Stander and Heyns (2002[a]) for their work on gearboxes operating under fluctuation load conditions. The details of the accelerated gear life test rig are presented in Section 4.2.1.

Figure 2.4 (a) shows a plot of the once per revolution pulse signal that would typically be obtained from a shaft encoder to compute the period of each shaft rotation. This signal was measured over a period of 32 seconds in which time the shaft rotated 165 times therefore the shaft encoder gives 165 pulses. Each pulse represents the start of a new gear rotation. This signal is used to synchronise the vibration data measured from the gearbox casing with the rotation of the gear. In Figure 2.4 (a) only 2 seconds of the pulse signal are shown to enhance clarity. Figure 2.4 (b) shows the time domain representation of the measured acceleration signal from the gearbox casing. The acceleration signal was measured in the vertical direction over a period of 32 seconds at a sampling frequency of 51200 Hz, but to enhance the clarity of the figure only 0.5 seconds of measured vibration are shown. Figure 2.4 (c) shows the vibration signals produced by five rotations superimposed on the time domain average that is calculated from 160 gear rotations. From this plot it is observed that the amplitude of the time domain average is less than that of the original signals. This is because the broad-spectrum noise component $e(t)$ has been filtered out through the time domain averaging

process. Another way of looking at this is by observing the RMS value of the TDA as a function of the number of input rotations. Figure 2.4 (d) is a plot of the RMS of the TDA against the number of signals that are used to compute the TDA (number of averages). From this plot it is observed that the RMS value of the TDA decreases as the number of inputs (gear rotations) that are used to calculate the TDA is increased. This is because the non-synchronous component of the gear vibration is filtered out as the number of inputs is increased.

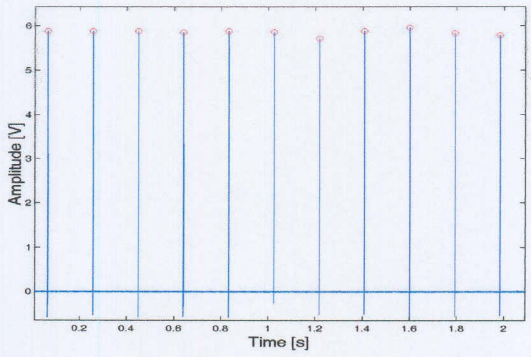


Figure 2.4 (a) One pulse per revolution shaft encoder signal used to synchronise the gear vibration with the gear rotation.

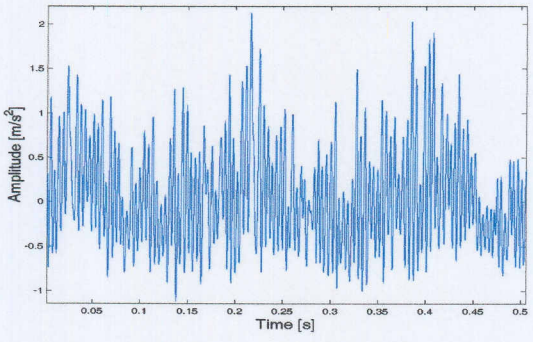


Figure 2.4 (b) Measured gear vibration signal over 0.5 seconds.

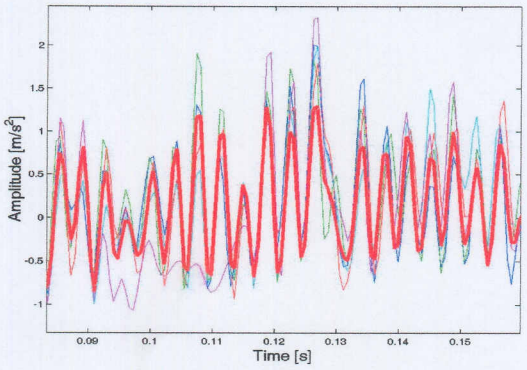


Figure 2.4 (c) Five rotation synchronised gear vibration signals superimposed on the TDA obtained after 160 gear rotations (red signal).

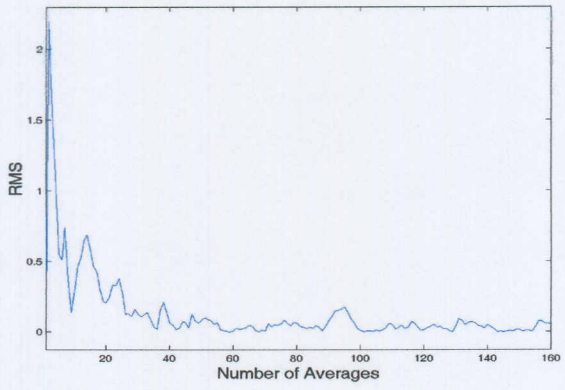


Figure 2.4 (d) TDA RMS vs. number of averages used to calculate TDA

Figure 2.5 illustrates the broad band filtering capabilities of TDA. The FFT spectrum of the TDA after 160 shaft rotations superimposed on the FFT spectrum of the original gear vibration signal. It is observed from Figure 2.5 that the noise content of the original signal $e(t)$ and the frequency content that is asynchronous to the rotation of the gear of interest have been filtered out. Only the gear mesh frequency (GMF) and its sidebands (SB.1 and SB.2) remain in the spectrum of the TDA. It is also observed that the amplitude of the GMF and SB.2 has increased. The amplitude of the spectrum of the TDA is generally less than the amplitude of the spectrum of the original signal at the frequencies that are not synchronous to the rotation of the gear of interest. These observations confirm the fact that calculating the TDA by direct averaging isolates the vibration produced by the rotation of a specific component, therefore the TDA calculated by direct averaging can be utilised to improve the diagnostic capability of a condition monitoring system.

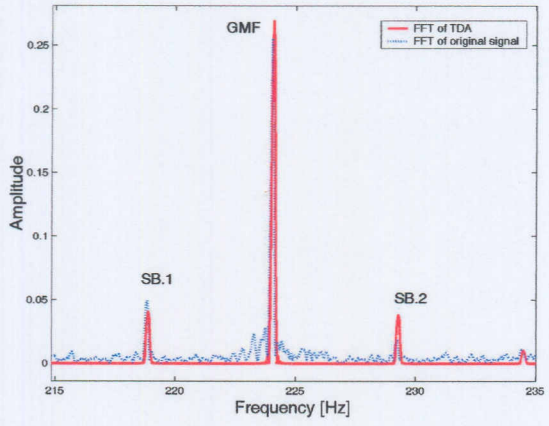


Figure 2.5 FFT of TDA after 160 gear rotation superimposed on the FFT the original gear vibration measured from the gearbox casing.

Another important observation is that calculating the TDA by direct averaging has filtered over an overlapping frequencies because it removed the noise over the entire frequency spectrum, while retaining all the frequency content that is related to the gear interest, in this case, the gear mesh frequency and its side bands. This capability gives calculating the TDA by direct averaging an advantage over other TDA models and linear filters, which can only retain or reject specific frequency bands.

2.3 Conclusion

In this chapter different approaches for calculating the TDA are presented. It is demonstrated that the comb filter model for time domain averaging is suitable for extracting specific frequencies and their harmonics from a signal when the period of the signal is known and constant. This model is, however, not effective when the frequency content of the noise coincides with that of the required signal as is commonly the case in many industrial applications. For gearboxes, a more suitable model for calculating the TDA is direct averaging. It is demonstrated that the direct averaging approach can filter out broadband noise over the entire spectrum of the signal leaving only the vibration content that is synchronous with the rotation of the gear of interest. This capability gives calculating the TDA by direct averaging an advantage over other TDA models and linear filters; therefore, in this study the TDA is calculated by direct averaging. Calculating the TDA by direct averaging requires an enormous amount of vibration data, and therefore, would still remain the main bottleneck in the development of an online gear condition monitoring system that utilises the TDA calculated by direct averaging to enhance its diagnostic capability. The TDA models developed later in this work seek to reduce the amount of vibration data that is required to calculating the TDA by direct averaging while retaining all the properties of the TDA.

Chapter 3

Artificial neural networks and support vector machines

3.1 Introduction

Neural networks have found extensive use in many industrial applications for pattern recognition. In this thesis the aim is to apply the ANNs and SVMs in pattern recognition as a predictive tool. The idea is to train ANNs and SVMs to predict the ensemble average of a large number of rotation synchronised gear vibration signals using only a portion of the total number of signals. This is in essence non-linear mapping between the input and output space, a task that ANNs and SVMs have handled with success (Fidêncio et al, 2002; Gunn, 1988). The averaging of the rotation synchronised vibration signals is called time domain averaging as discussed in Chapter 2. It was also demonstrated that calculating the TDA by direct averaging can filter out broadband noise over the entire spectrum of the signal leaving only the vibration content of interest. The ANNs and SVMs mapping should therefore retain the non-linear filtering achieved in the frequency domain by the TDA calculated by direct averaging. Before a synchronous time domain averaging model can be developed for gear vibration using ANNs and SVMs, it is essential to have thorough understanding of their underlying mathematics. In this chapter, the theory of Multi-layer Perceptron (MLP) networks, Radial Basis Function (RBF) networks (Bishop, 1995) and Support Vector Machines (SVM) (Vapnik, 1995; Vapnik et al., 1997; Gunn, 1998) in the light of this work is presented. This chapter also presents simulation results based on a preliminary study conducted on a data set from an accelerated gear life test rig. The preliminary study is used to investigate the suitability of these methods for application in the development of a time-domain averaging model for gear vibration.

3.2 Artificial neural networks

In this work neural networks are viewed as parameterised non-linear mapping of input data to the output data. Learning algorithms are viewed as methods for finding parameter values that look probable in the light of the data. The learning process occurs

by training the network through supervised learning. Supervised learning is the case where the input data set ($\boldsymbol{X}$) and the output data set ($\boldsymbol{Y}$) are both known and neural networks are used to approximate the functional mapping between the two data sets. In this section the theory and application of MLP and RBF networks formulations are presented. Simulations to assess the suitability of each of these formulations for use in the development of a synchronous time domain averaging model using a data set from the accelerated gear life test rig (Stander and Heyns, 2002[b]) are presented.

3.2.1 Multi-layer perceptron

The MLP provides a distributed representation with respect to the input space due to the cross-coupling between hidden units. In this study, the MLP architecture contains a hyperbolic tangent basis function in the hidden units and linear basis functions in the output units (Bishop, 1995). A schematic illustration of a 2-layer MLP network is shown in Figure 3.1.

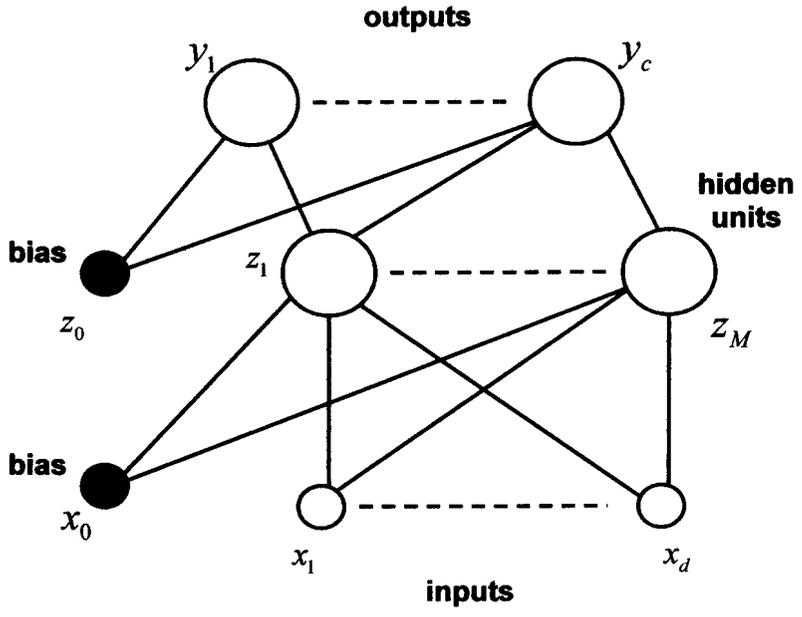


Figure 3.1 Feed-forward network with two layers of adaptive weights (Bishop, 1995).

The MLP network architecture in Figure 3.1 contains hidden units, output units, and one hidden layer. The bias parameters in the first layer are shown as weights from an extra input having a fixed value of $x_0 = 1$. The bias parameters in the second layer are shown as weights from an extra hidden unit, with the activation fixed at $z_0 = 1$. The model in

Figure 3.1 is able to take into account the intrinsic dimensionality of the data. Models of this form can approximate any continuous function to arbitrary accuracy if the number of hidden units M is sufficiently large. The size of a MLP network can be expanded by considering several layers but this is not necessary because it has been demonstrated through the Universal Approximation Theorem (Haykin, 1999) that a two-layered architecture is adequate for the multi-layer perceptron. As a result of this theorem, in this study a two-layered network shown in Figure 3.1 is chosen.

The output of the MLP network represented in Figure 3.1 is given by Equation 3.1.

$$y_k = f_{outer}\left(\sum_{j=0}^{M} w_{kj}^{(2)} f_{inner}\left(\sum_{i=0}^{d} w_{kj}^{(1)} x_i\right) + w_{k0}^{(2)}\right) \tag{3.1}$$

where f_{outer} and f_{inner} are activation functions, $w_{ji}^{(1)}$ denotes a weight in the first layer, going from input i to hidden unit j, $w_{k0}^{(2)}$ is the bias for the hidden unit k and $w_{kj}^{(2)}$ denotes a weight in the second layer. In this work f_{inner} is a hyperbolic tangent function "tanh" and f_{outer} is linear. The linear activation function is defined by Equation (3.2) and it maps the interval (-∞,∞) onto the interval (-∞,∞).

$$f_{outer}(v) = v \tag{3.2}$$

The hyperbolic tangent function is defined by

$$f_{inner}(v) = \tanh(v) = \frac{e^{v} - e^{-v}}{e^{v} + e^{-v}} \tag{3.3}$$

and it maps the interval (-∞,∞) onto the interval (-1,1). Figure 3.2 below shows the two activation functions used in this study.

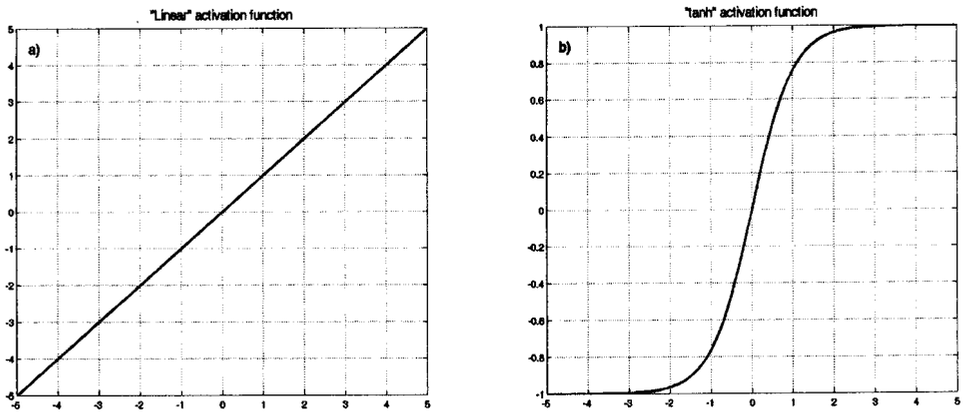


Figure 3.2 (a) Plot of the linear activation function given by equation (3.2). (b) Plot of the 'tanh' activation function given by Equation (3.3).

Training the neural network is achieved by calculating the weights in Equations (3.1). There are two principal approaches that can be used to train neural networks. These methods are the maximum-likelihood approach and the Bayesian approach (Bishop, 1995). In maximum-likelihood training, optimisation methods are used to identify a set of parameter values that maximises the ability of a network to predict the output whenever presented with the input data. The Bayesian method uses Bayes's theorem (Bishop, 1995) to identify the probability distribution of weights in the light of the training data that are initially set to some prior distribution. The maximum-likelihood method may be viewed as a special case of the Bayesian method. In this work the maximum-likelihood method is implemented for computational efficiency.

3.2.2 Maximum-likelihood-based cost function

In the maximum-likelihood approach an optimisation procedure is used to identify the weights and biases of the neural networks in Equation (3.1). A cost function is chosen in order to use the optimisation technique. A cost function is a mathematical representation of the overall objective of the problem (Marwala, 2001). In this work, the overall objective is to identify a set of neural network weights, that can map the rotation synchronised gear vibration signal (input) to the ensemble average of the rotation synchronised gear vibration signals (TDA obtained from the k signals). The mapping should use less rotation synchronised gear vibration signals than would otherwise be required to calculate the TDA using direct averaging. In other words,

predicting the TDA of a rotating gear using only a fraction of the number of rotation synchronised gear vibration signals that would be used in the direct averaging approach.

If the training set $D=\{X_k,t_k\}_{k=1}^{N}$ is used and assuming that the targets t_k are sampled independently given the inputs X_k and the weight parameters, w_{kj}, the cost function, E, may be written as

$$E=\sum_n\sum_k\{t_{nk}-y_{nk}\}^2+\frac{\alpha}{2}\sum_j^w w^2 \tag{3.4}$$

where, n is the index for the training pattern and k is the index for the output units. The first term in Equation (3.4) is the sum-of-square-of-errors cost function, which tends to give similar absolute errors for each pattern. This results in poor performance on target values of small magnitude. The other cost function that has been used is the cross-entropy cost function (Hopfield, 1987; Hinton, 1987). Minimisation of the cross-entropy cost function tends to give the same relative errors for small and large targets. The cross-entropy cost function plus the weight decay regularisation parameters may be written as follows:

$$E=-\sum_n\sum_k\{t_{nk}\ln(y_{nk})+(1-t_{nk})\ln(1-y_{nk})\}+\frac{\alpha}{2}\sum_j^w w_j^2 \tag{3.5}$$

The cost function in Equation (3.5) has been found to be suited for classification problems while the one in Equation (3.4) has been found to be suited for regression problems (Bishop, 1995). Since the application in this work is a type of regression Equation (3.4) is used.

3.2.3 Regularisation

The second term in Equation (3.4) is the regularisation parameter. The regularisation parameter in Equation (3.4) penalises large weights and ensures that the mapping function is smooth (Vapnik, 1995). This regularisation parameter is called the weight decay and its coefficient, α, determines the relative contribution of the regularisation term on the training error. The inclusion of the regularisation parameter has been found to give significant improvements in network generalisation (Hinton, 1987).

In neural networks, to produce an over-fitted mapping with regions of large curvature requires large weights. The weight decay regularisation penalises large weights thereby encouraging the weights to be small and avoiding an over-fitted mapping between the inputs and the outputs. If α is too high then the regularisation parameter over-smoothes the network weights and as result giving inaccurate results. If α is too small then the effect of the regularisation parameter is negligible and unless other measures that control the complexity of the model, such as the early stopping method (Bishop 1995) are implemented, the trained network becomes too complex and thus performs poorly on validation sets.

3.2.4 MLP network training

Before minimisation of the cost function is performed, the network architecture needs to be constructed by choosing the number of hidden units, M. If M is too small, the neural network will be insufficiently flexible and will give poor generalisation of the data because of high bias. However, if M is too large, the neural network will be unnecessarily flexible and will give poor generalisation due to a phenomenon known as overfitting caused by high variance (Geman et al.,1992). The weights (w_i) and biases in the hidden layers are varied using optimisation methods until the cost function is minimised. Gradient descent methods are implemented and the gradient of the cost function is calculated using the back-propagation method (Bishop, 1995). The details of the back-propagation method are found in Appendix B. In this work it was decided to use the Scaled Conjugate Gradient (SCG) method over Conjugate Gradient (CG). This choice was made because SCG method is more computational efficient than CG while retaining the essential advantages of the CG method (Haykin, 1999; Marwala, 2001). The details of these optimisation methods are explained in Appendix C.

3.2.5 MLP simulation results from a preliminary investigation using data from the accelerated gear life test rig

This section seeks to validate the suitability of the MLP network for use in synchronous TDA model using gear vibration data from the accelerated gear life test rig. This was done in the following steps.

- Data pre-processing. The acceleration signal measured from the gearbox casing was synchronised with the rotation of the gear using the one pulse per revolution

shaft encoder signal. This resulted in 160 rotation synchronised gear vibration signals. Each signal contained 8192 sample points. The rotation synchronised gear vibration signals were resampled to 1024 points per signal to reduce computational load. This results in an input space X_k of dimension (160×1024). The target t_k is the TDA of the gear vibration calculated using the direct averaging approach for 160 gear rotations. The dimensions of the target are (1×1024).

- Selecting type of network. A two-layer network was selected because the Universal approximation Theorem (Haykin, 1999) states that a two-layered network is sufficient for mapping data of arbitrary complexity (Marwala, 2001).
- Selecting number of hidden units. The number of hidden units was chosen between 3 and 15 and the one that resulted in the least square errors when simulating with unseen validation sets was selected. In this application 10 hidden units were selected because they resulted in a small error without severe computational penalties.
- Selected activation functions. The hyperbolic tangent function was selected as the inner activation function and a linear function was selected as the outer activation function.
- Type of optimisation technique used. The scaled conjugate gradient optimisation technique was used to optimise the cost function because of its computational efficiency.
- Selecting number of inputs. The number of inputs was chosen between 1 and 50 and the one that resulted in the least square errors when simulating with unseen validation sets was selected.
- The regularisation coefficient, α, was selected by trial and error, starting at a value of $\alpha = 0$, and increasing α sequentially in steps $\alpha = 0.1$ until satisfactory smoothness of the predicted result was obtained from simulations with unseen data. In this work $\alpha = 1.5$ was found to be most suitable.

The following plots show the performance of MLP neural network on a preliminary study conducted using vibration data from the accelerated gear life test rig (Stander and Heyns, 2002). Figure 3.3 shows MLP simulation results for a network of 10 hidden units and 40 input vectors (40 rotation synchronised vibration signals), each signal with

signal has 1024 points resulting in an input space of (40 □1024). From this simulation it is clearly evident that this MLP network architecture is suitable for use in a synchronous TDA model in that it correctly predicts the target (time domain average after 160 gear rotations with only 40 gear rotation signals).

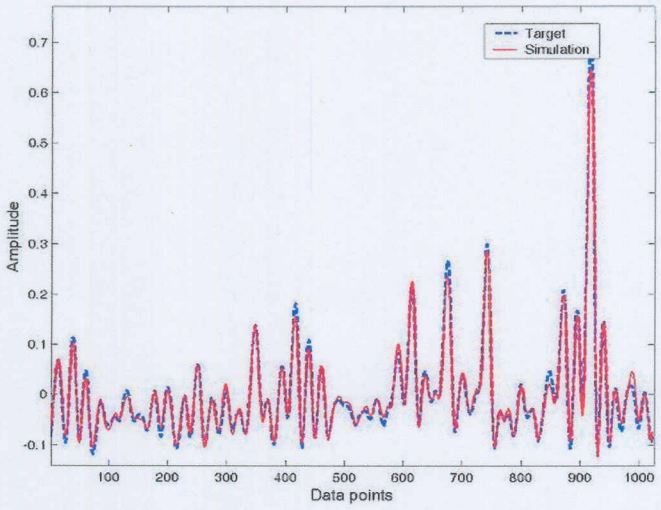


Figure 3.3 MLP simulation results for network with 40 inputs and 10 hidden units superimposed on TDA calculated by direct averaging.

Figure 3.4 shows the simulation results of validation sets as a function of the number of inputs. It is observed that the RMS error stabilises after 40 input vectors, therefore, 40 inputs are selected as the optimum number of inputs, therefore the amount of data that is required to calculate the TDA is reduces by 75 percent.

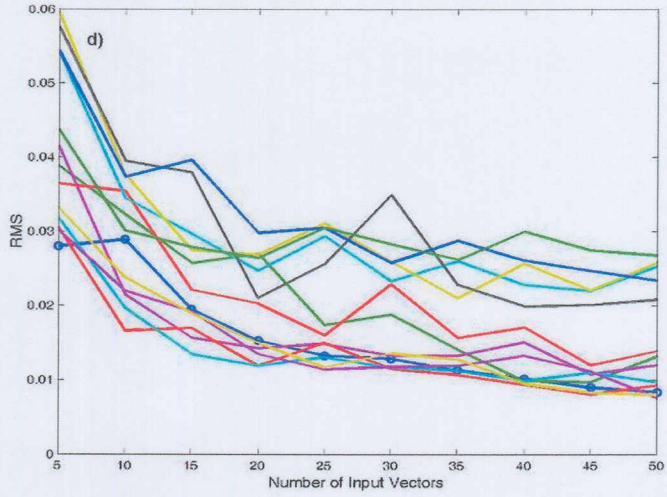


Figure 3.4 RMS of MLP simulation results for network with 10 hidden units vs. Number of input signals.

3.2.6 Radial basis functions

The radial basis function has its origin in techniques for performing exact interpolation of a set of data points in a multi dimensional space (Powell, 1987). The exact interpolation problem requires every input vector to be mapped exactly to the corresponding vector and will be used in the discussion of Radial Basis Function (RBF) network (Bishop, 1995). The RBF neural network can be obtained by introducing a couple of modifications to the exact interpolation process (Broomhead, 1988; Moody and Darken, 1989). The RBF neural networks provide a smooth interpolating function for which the number of basis functions is determined by the complexity of the mapping to be represented rather than the data set as in exact interpolation. In this work, the objective is to identify a set of basis functions, that can map the rotation synchronised gear vibration signal (input) to the ensemble average of the rotation synchronised gear vibration signals (TDA obtained from the k signals). The mapping should use less rotation synchronised gear vibration signals than would otherwise be required to calculate the TDA using direct averaging.

The RBF neural network mapping is given by given

$$y_k(\mathbf{x}) = \sum_{j=1}^{M} \omega_{kj}\phi_j(\mathbf{x}) + \omega_{k0} \tag{3.6}$$

where ω_{k0} are the biases, ω_{kj} are the basis function weights, $\mathbf{x}$ is the d-dimensional input vector and $\phi_j(\cdot)$ is the jth basis function. Several forms of basis functions have been considered, the most common being the Gaussian given by

$$\phi_j(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}_j\|^2}{2\sigma_j^2}\right) \tag{3.7}$$

where $\mathbf{x}$ is the d-dimensional input vector with elements x_i , and $\boldsymbol{\mu}_j$ is the vector determining the centre of the basis function ϕ_j and has elements μ_{ji}. For further detail on the selection of the basis function centres μ_{ji} see Bishop (1995). The parameter σ controls the smoothness properties of the basis function. The Gaussian interpolation

function is a localised basis function with the property that $\phi \to 0$ as $|x| \to \infty$. Another basis function that shares the properties of the Gaussian is given by

$$\phi_j(\mathbf{x}) = \left(\mathbf{x}^2 + \sigma^2\right)^{-\alpha} \tag{3.8}$$

It is, however, not necessary for this function to be localised. Other possible choices are the thin-plate spline function given by

$$\phi_j(\mathbf{x}) = \mathbf{x}^2 \ln(\mathbf{x}), \tag{3.9}$$

the function

$$\phi_j(\mathbf{x}) = \mathbf{x}^4 \ln(\mathbf{x}), \tag{3.10}$$

the function

$$\phi_j(\mathbf{x}) = \left(\mathbf{x}^2 + \sigma^2\right)^{\beta}, \qquad 0 < \beta < 1, \tag{3.11}$$

which for $\beta = 1/2$ is known as the multi-quadratic function, the cubic function

$$\phi_j(\mathbf{x}) = \mathbf{x}^3 \tag{3.12}$$

and the 'linear' function

$$\phi_j(\mathbf{x}) = \mathbf{x} \tag{3.13}$$

which all have the property that $\phi \to \infty$ as $x \to \infty$. The RBF network is represented by the diagram in Figure 3.3.

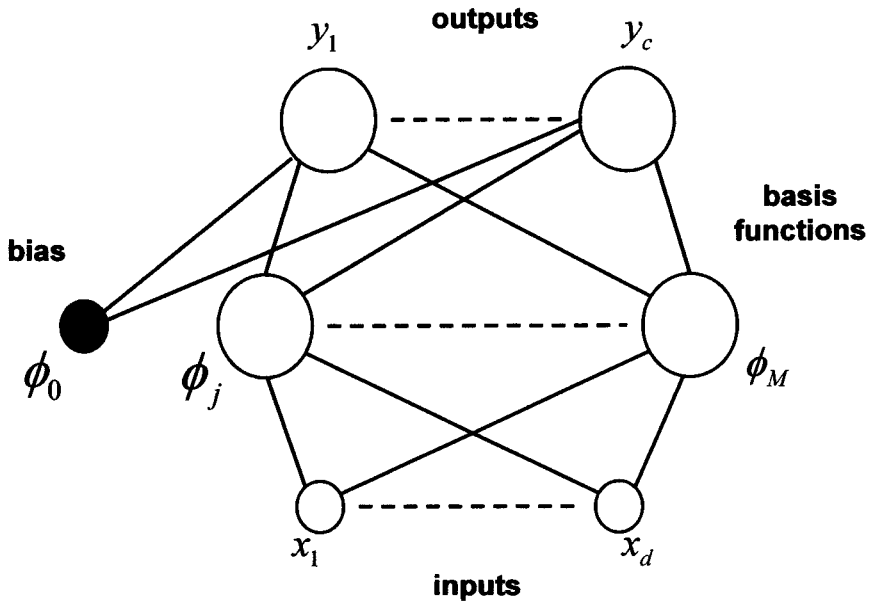


Figure 3.5 Architecture of a radial basis function network (Bishop, 1995).

In Figure 3.5 each basis function acts like a hidden unit. The lines connecting basis function ϕ_j to the inputs represent the corresponding element μ_{ji} of the vector μ_j. The weights ω_{kj} are shown as lines from the basis functions to the output units, and the biases are shown as weights from an extra 'basis function' ϕ_0 whose output is fixed to 1. The Gaussian radial basis function considered above can be generalised to allow for arbitrary covariance matrix $\sum_j$ by changing the form of the basis function to

$$\phi_j(\mathbf{x}) = \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_j)^T \sum_j^{-1}(\mathbf{x}-\mu_j)\right\}. \tag{3.14}$$

Since the covariance matrices $\sum_j$ are symmetric, each basis function has $d(d+30)/2$ independent adjustable parameters (where d is the dimensionality of the input space) (Bishop, 1995).

3.2.7 RBF network training

The training of the radial basis function takes place in two stages. In the first stage the input data set $\mathbf{x}^n$ alone is used to determine the parameters of the (μ_j and σ_j for the spherical radial basis function). After the first stage of training the basis functions are

then kept fixed and the second layer of weights are found in the second training phase presented below. When the bias parameters in Equation (3.6) are absorbed into the weights the resulting equation is

$$y_k(\mathbf{x}) = \sum_{j=0}^{M} \omega_{kj} \phi_j(\mathbf{x}) \tag{3.15}$$

where ϕ_0 is an extra 'basis function' with activation value fixed at $\phi_0 = 1$. In matrix form this equation is

$$\mathbf{y}(\mathbf{x}) = \mathbf{W}\boldsymbol{\varphi} \tag{3.16}$$

where $\mathbf{W} = (\omega_{kj})$ and $\boldsymbol{\varphi} = (\phi_j)$. Since the basis functions are considered fixed, the network is equivalent to a single–layer network that can be optimised by minimisation of a suitable error function. The sum-of-square error function is given by

$$E = \frac{1}{2} \sum_n \sum_k \left\{ y_k(\mathbf{x}^n) - t_k^n \right\}^2 \tag{3.17}$$

where t_k^n is the target value for the unit k when the network is presented with input vector $\mathbf{x}^n$. Since the error function is a quadratic function of the weights, its minimum can be found in terms of the solution of a set of linear equations. The weights are determined by the linear equation

$$\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \mathbf{W}^{\mathrm{T}} = \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{T} \tag{3.18}$$

where $(\mathbf{T})_{nk} = t_k^n$ and $(\boldsymbol{\Phi})_{nj} = \phi_j(\mathrm{x}^n)$. The formal solution of the weights is given by

$$\mathbf{W}^{\mathrm{T}} = \boldsymbol{\Phi}^{\dagger} \mathbf{T}, \qquad \text{where } \boldsymbol{\Phi}^{\dagger} = \left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}}. \tag{3.19}$$

$\boldsymbol{\Phi}^{\dagger}$ is the pseudo-inverse of $\boldsymbol{\Phi}$. In practice Equations (3.18) are solved using singular value decomposition, to avoid problems due to possible ill conditioning of the matrix $\boldsymbol{\Phi}$.

For regression the basis function parameters can be found by treating the basis function centres and widths, along with the second-layer weights, as adaptive parameters to be determined by the minimisation of an error function. For the case of the sum-of-squares error, and spherical Gaussian basis functions Equation (3.7), the following expressions are obtained for the derivatives of the error function with respect to the basis function parameters

$$\frac{\partial E}{\partial \sigma_j} = \sum_n \sum_k \left\{ y_k\left(\mathrm{x}^n\right) - t_k^n \right\} \omega_{kj} \exp\left(-\frac{\left\| \mathrm{x}^n - \mu_j \right\|^2}{2\sigma_j^2} \right) \frac{\left\| \mathrm{x}^n - \mu_j \right\|^2}{\sigma_j^3} \tag{3.20}$$

$$\frac{\partial E}{\partial \mu_{ji}} = \sum_n \sum_k \left\{ y_k\left(\mathrm{x}^n\right) - t_k^n \right\} \omega_{kj} \exp\left(-\frac{\left\| \mathrm{x}^n - \mu_j \right\|^2}{2\sigma_j^2} \right) \frac{\left\| x_i^n - \mu_{ji} \right\|^2}{\sigma_j^2} \tag{3.21}$$

where μ_{ji} denotes the *i*th component of μ_j. These expressions for the derivative are used in conjunction with standard optimisation strategies. The setting of the basis function parameters by supervised learning represents a non-linear optimisation problem that is computationally expensive and may be prone to finding local minima in the error function but if the relevant basis functions are identified the training procedures can be significantly speeded up (Bishop, 1995).

3.2.8 RBF simulation results from a preliminary investigation using data from the accelerated gear life test rig

In this section the suitability of the RBF neural networks for use in synchronous TDA model is assessed using gear vibration data from the accelerated gear life test rig. To achieve this the following steps were followed.

- Data pre-processing. The data was pre-processed as explained in Section 3.2.5 resulting in an input space X_k of dimension (160×024) and a t_k of dimensions (1×1024).
- Selecting number of basis. The number of number of basis was chosen between 1 and 10 and the one that resulted in the least square errors when simulating with unseen validation sets was selected. In this application 5 number of basis were

selected because they resulted in a small error without severe computational penalties.

- Selected type of basis functions. The activation functions that were investigated for setting the radial basis function structures were: the radially symmetric Gaussian function, the thin plate splines 'tps' $\phi(x) = x^2 \ln(x)$, and the $\phi(x) = x^4 \ln(x)$ activation functions. The output error function was defined as linear. The type that resulted in the least square errors when simulating with unseen validation sets was selected was the thin plate splines therefore it was implemented in this study.
- Selecting number of inputs. The number of inputs was chosen between 1 and 40 and the one that resulted in the least square errors when simulating with unseen validation sets was selected. In this application 40 inputs resulted in the smallest square errors, therefore resulting in a data reduction of 75 percent for calculating the TDA.

Figures 3.6 and 3.7 show the performance of RBF neural network on a preliminary study conducted using vibration data from the accelerated gear life test rig.

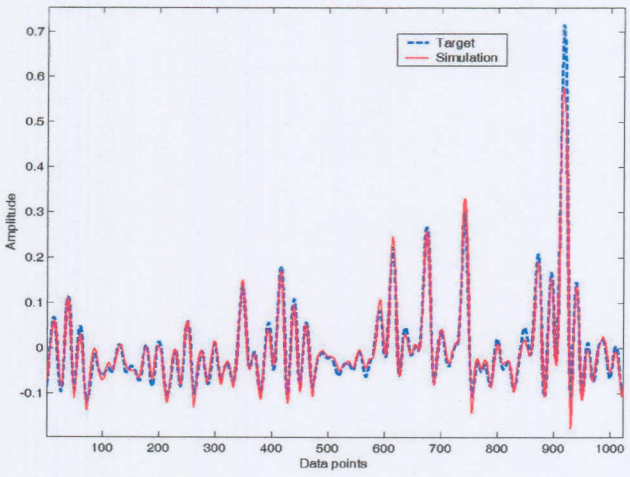


Figure 3.6 RBF simulation results for network with 40 inputs and 5 basis functions superimposed on TDA obtained by direct averaging.

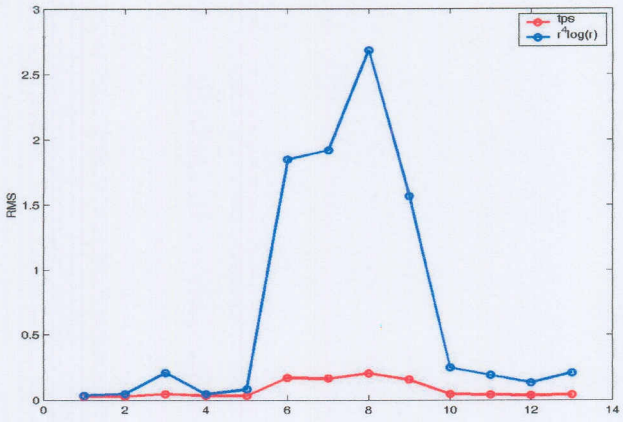


Figure 3.7 Performance of the thin plate spline 'tps' and the $\varphi = \mathbf{x}^4 \ln(\mathbf{x})$ basis functions on 13 validation sets.

Figure 3.6 shows the simulation results of a RBF neural network with 5 basis functions and 40 input vectors simulated using an unseen validation data set. From this simulation it is observed that the RBF network can be used to predict the time domain average of the gearbox signal using less data. Figure 3.7 shows the performance of thin-plate spline'tps' $\phi(x) = x^2 \ln(x)$, and the $\phi(x) = x^4 \ln(x)$, using the validation sets. From this simulation it is observed that the 'tps' activation function performs better than the $\phi(x) = x^4 \ln(x)$ activation function because it results in a constant prediction error for all the validation sets. This is because the 'tps' activation generalises well to all the validation sets as opposed to the $\phi(x) = x^4 \ln(x)$ activation function that does not generalise resulting in a peak in prediction error between validation set 5 and validation set 10. This is because the validation sets that were used during simulation were taken from different stages of the gear life and had different vibration signatures representing the progression in the level of damage. In this work 'tps' activation function is therefore selected for all RBF analysis.

3.3 Support vector machines

Traditional neural network approaches have suffered difficulties with generalisation, producing models that can overfit the data. This is a consequence of the optimisation algorithms used for parameter selection and the statistical measures used to select the 'best' model (Gunn, 1998). The foundations of Support Vector Machines (SVMs) have

been developed by Vapnik (1995) and are gaining popularity due to many attractive features, and promising empirical performance. The SVM formulation embodies the Structural Risk Minimisation (SRM) principle, which has been shown to be superior (Gunn et al., 1997), to traditional Empirical Risk Minimisation (ERM) principle, employed by conventional neural networks. SRM minimises an upper bound on the expected risk, as opposed to ERM that minimises the error on the training data. It is this difference that equips SVMs with a greater ability to generalise, which is the goal in statistical learning. SVMs were originally developed for classification problems, but recently have been extended to regression problems (Vapnik et al., 1997). In this work we are interested in the regression properties of support vector machines and therefore the term support vector machines refers to SVMs applied to regression problems. The objective is to map the rotation synchronised gear vibration signal (input) to the ensemble average of the rotation synchronised gear vibration signals (TDA obtained from the k signals) using less rotation synchronised gear vibration signals than would otherwise be required to calculate the TDA using direct averaging.

3.3.1 Linear regression

When considering a simple linear regression problem of approximating a set of data,

$$D=\left\{\left(x^1,y^1\right),\mathrm{K},\left(x^l,y^l\right)\right\},\qquad x\in ¡^n, y\in ¡, \tag{3.22}$$

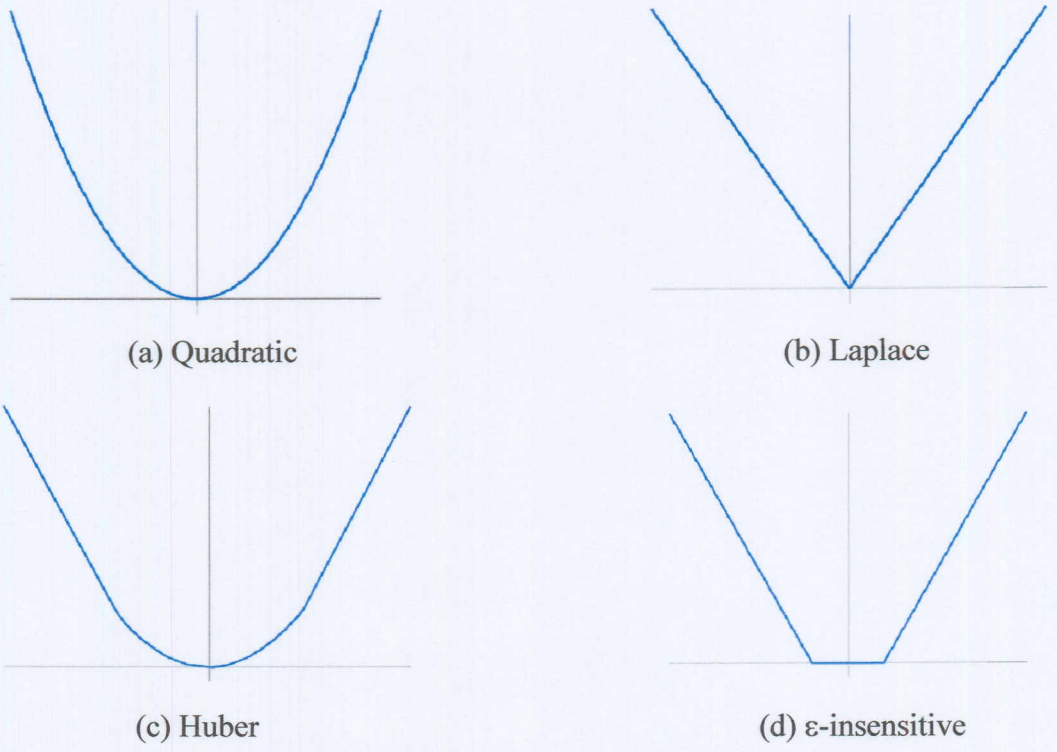
with a linear function ,

$$f\left(x\right)=\left\langle w,x\right\rangle+b. \tag{3.23}$$

The optimal regression function is given by the minimum of the functional,

$$\Phi\left(w,\xi\right)=\frac{1}{2}\|w\|^2+C\sum_i\left(\xi_i^-+\xi_i^+\right), \tag{3.24}$$

where C is a pre-specified value, w are the weights and ξ^-,ξ^+ are slack variables representing the lower and upper constraints on the outputs of the system. When support vector machines are applied to regression problems loss functions that include a distance measure are used. Figure 3.8 below shows some of the possible loss functions.



(a) Quadratic (b) Laplace

(c) Huber (d) ε-insensitive

Figure 3.8 Possible loss functions support vector regression.

The loss function in Figure 3.8 (a) corresponds to the conventional least squares error criterion. The loss function in Figure 3.8 (b) is a Laplacian loss function that is less sensitive to outliers than the quadratic loss function. The Huber loss function in Figure 3.8 (c) is a robust loss function with optimal properties when the underlying distribution of the data is not known. These three loss functions produce no sparseness in the support vector. To address this issue Vapnik proposed the loss function in Figure 3.8 (d) as an approximation to the Huber loss function. The ε-insensitive enables the sparse set of support vectors to be obtained.

Using the ε-insensitive loss function in Figure 3.8 (d),

$$L_{\epsilon}(y)=\begin{cases}0 & \text{for} \quad |f(\mathrm{x})-y|<\varepsilon \\ |f(\mathrm{x})-y|-\varepsilon & \text{Otherwise}\end{cases} \tag{3.25}$$

the solution to the optimal regression function in Equation (3.24) is given by,

$$\max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle + \sum_{i=1}^{l}\alpha_i(y_i-\epsilon)-\alpha_i^*(y_i+\epsilon) \tag{3.26}$$

or alternatively,

$$\bar{\alpha},\bar{\alpha}^* = \arg\min_{\alpha,\alpha^*} -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle - \sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i + \sum_{i=1}^{l}(\alpha_i+\alpha_i^*)\epsilon \tag{3.27}$$

with constraints,

$$0 \le \alpha_i,\alpha_i^* \le C, \qquad i = 1,\mathrm{K}\ ,l$$
$$\sum_{i=1}^{l}(\alpha_i-\alpha_i^*) = 0. \tag{3.28}$$

Solving Equation (3.26) with constraints in Equation (3.28) determines the Lagrange multipliers, α,α^* and the regression function is given by Equation (3.23), where

$$\bar{w} = \sum_{i=1}^{l}(\alpha_i-\alpha_i^*)x_i$$
$$\bar{b} = -\frac{1}{2}\langle \bar{w},(x_r+x_s)\rangle. \tag{3.29}$$

The Karush-Kuhn-Tucker (KKT) conditions (Vapnik, 1995) that are satisfied by the solution are,

$$\bar{\alpha}_i\bar{\alpha}_i^* = 0, \qquad i = 1,\mathrm{K}\ ,l. \tag{3.30}$$

Therefore the support vectors are points where exactly one of the Lagrange multipliers are greater than zero. When $\epsilon = 0$, we get the L_1 loss function and the optimisation problem is simplified to

$$\min_{\beta} \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\beta_i\beta_j\langle x_i, x_j\rangle - \sum_{i=1}^{l}\beta_i y_i \tag{3.31}$$

with constraints,

$$\begin{aligned} &-C \le \beta_i \le C, \qquad i = 1,\text{K}\ ,l \\ &\sum_{i=1}^{l}\beta_i = 0, \end{aligned} \tag{3.32}$$

and the regression function is given by Equation (3.23), where

$$\begin{aligned} \bar{w} &= \sum_{i=1}^{l}\beta_i x_i \\ \bar{b} &= \frac{1}{2}\langle \bar{w}, (x_r + x_s)\rangle. \end{aligned} \tag{3.33}$$

Details for the implementation of the other three loss functions are presented in Appendix D.

3.3.2 Non-linear regression

In cases where non-linear regression is required, non-linear mapping is used to map the data to a higher dimensional feature space where linear regression is performed. The kernel approach is employed to address the curse of dimensionality. The non-linear support vector regression solution, using the ε-insensitive loss function is given by

$$\begin{aligned} \max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} &\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i^*(y_i - \epsilon) - \alpha_i(y_i + \epsilon) \\ &-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j) \end{aligned} \tag{3.34}$$

with constraints,

$$\begin{aligned} &0 \le \alpha_i, \alpha_i^* \le C, \qquad i = 1,\text{K}\ ,l \\ &\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0. \end{aligned} \tag{3.35}$$

Solving Equation (3.34) with the constraints in Equation (3.35) determines the Lagrange multipliers, α, α^* and the regression function is given by

$$f(\mathrm{x}) = \sum_{i=1}^{l} (\bar{\alpha}_i - \bar{\alpha}_i^*) K(\mathrm{x}_i, \mathrm{x}) + \bar{b} \tag{3.36}$$

where

$$\begin{aligned} \langle \bar{w}, x \rangle &= \sum_{SVs} (\alpha_i - \alpha_i^*) K(x_i, x_j) \\ \bar{b} &= -\frac{1}{2} \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) (K(x_i, x_r) + K(x_i, x_s)). \end{aligned} \tag{3.37}$$

The optimisation criteria for other loss functions are similarly obtained by replacing the dot product with kernel functions. The ε-insensitive loss function is attractive because unlike the quadratic and Huber cost functions, where all the data points will be support vectors, the support vector solution can be sparse, therefore the ε-insensitive loss function was selected in this work. Different kernels were investigated for mapping the data to a higher dimensional feature space where linear regression is performed.

3.3.3 SVM simulation results from a preliminary investigation using data from the accelerated gear life test rig

In this section the suitability of SVMs for use in synchronous TDA model is assessed using gear vibration data from the accelerated gear life test rig. To achieve this the following steps were followed.

- Data pre-processing. The data was pre-processed as explained in Section 3.2.5 resulting in an input space X_k of dimension (160×1024) and a t_k of dimensions (1×1024).
- Selected type of Kernel functions. The Kernel functions that were investigated in this study were: Exponential Radial Basis Function (ERBF), the Gaussian Radial Basis function (RBF), the spline and the b-spline kernels and the one that resulted in the smallest error was selected. (see Appendix E).
- Selecting number of inputs. The number of inputs was chosen between 1 and 40 and the one that resulted in the least square errors when simulating with unseen

validation sets was selected. In this application 40 inputs resulted in a smallest square errors. This is effectively a data reduction of 75 percent.

Figures 3.9 to 3.11 show the performance SVMs on a preliminary study conducted using vibration data from the accelerated gear life test rig.

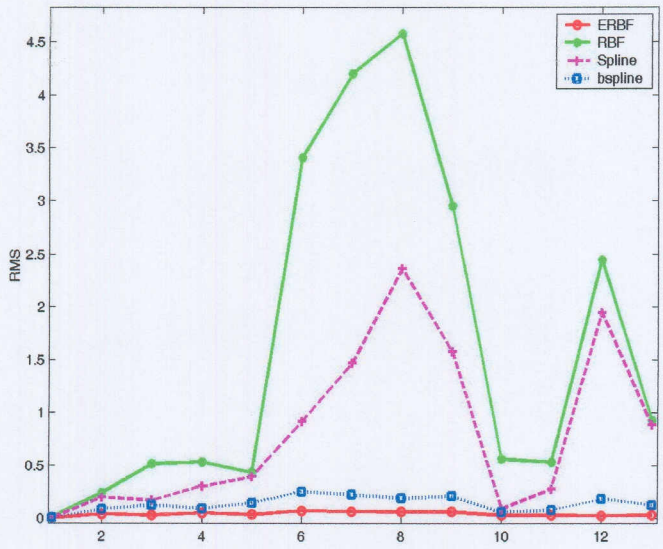


Figure 3.9 Performance of different Kernel function on 13 unseen validation sets

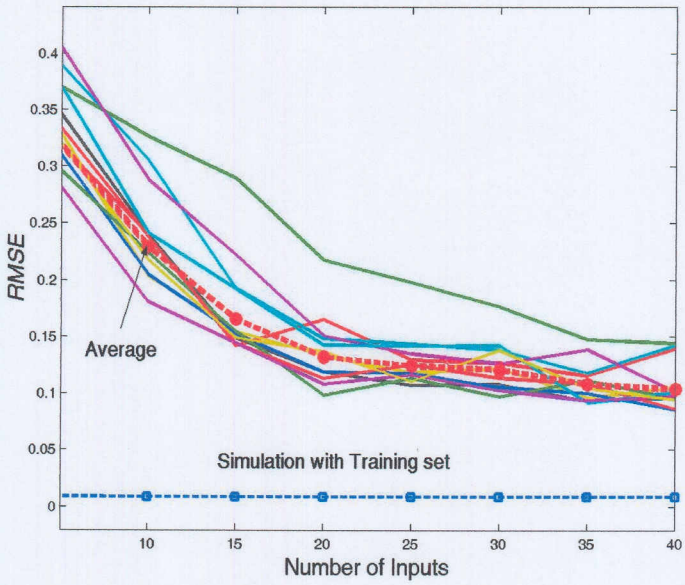


Figure 3.10 RMS of SVM simulation vs. Number of input signals

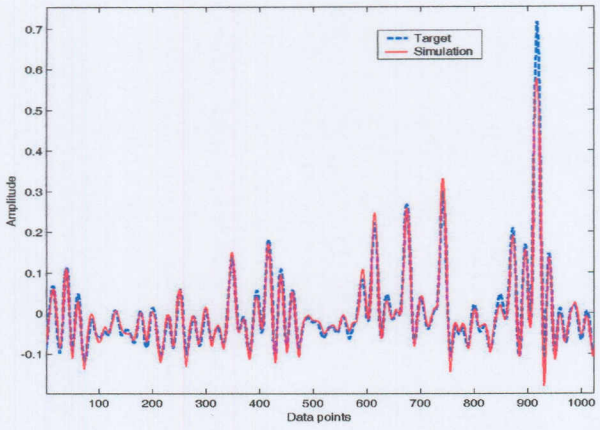


Figure 3.11 Prediction of SVM with 40 inputs and ERBF kernel superimposed on target

Figure 3.9 shows the performance of different Kernel functions on 13 validation sets from different stages of the gear life. It is observed that the ERBF Kernel gives the best mapping for all the validation sets. It is also observed that the RBF, spline and b-spline Kernel functions have a peak at validation set 8 and validation set 12. This is because the SVMs were trained with data from the early stages of the gear life. As the gear life progresses the vibration signature changes significantly. The peaks at validation set 8 and validation set 12 indicate that SVMs with RBF, spline and b-spline Kernel functions do no generalise well to the vibration signatures at these stages of gear life. Figure 3.10 shows the performance of SVM with an ERBF kernel as a function of number of input vectors on validation sets. From this plot it is observed that the RMS decreases as the number of inputs increases. Figure 3.12 shows the simulation results SVM with an ERBF Kernel and 40 unseen input signal superimposed on the TDA calculated using the direct time domain averaging approach. This plot indicates that SVMs can be effectively used to predict the TDA of the gearbox signal using less data than would be used to calculate the TDA using the direct averaging approach.

3.4 Conclusion

From the simulation results obtained in the preliminary study, it is observed that all three formulations can successfully map the input space to the TDA calculated by time domain averaging using only 25 percent of the input data. It is therefore concluded from the preliminary study that all three formulations are suitable for use in the development of a synchronous filter for time domain averaging of gear vibration data. Chapter 4 deals with the development issues for the synchronous filter.

Chapter 4

Development process for synchronous filter

4.1 Introduction

One of the main objectives for the development of a synchronous filter for gear vibration is to reduce the amount of gear vibration data that needs to be stored in the data acquisition system in order to calculate the time-domain average (synchronous average) of the gear vibration data. This space saving can bring us a step closer to the development of a truly online gear condition monitoring system that utilises time domain averaging to enhance the diagnostic capability. The reduction in the amount of data that needs to be stored in the data acquisition system allows for data acquisition and analysis to be executed simultaneously. In Chapter 3 the theory of MLP, RBF networks and SVMs in the light of this work was presented. Their suitability for use in a time domain averaging task was also assessed. In this chapter, two models for synchronous filtering of gear vibration are presented. The proposed models are implemented using each of the three mathematical formulations and their performances are compared. A detailed explanation of the experimental set-up is also presented.

4.2 ANNs and SVMs synchronous filtering model

In neural networks input space reduction is achieved by transforming the input data space into a lower dimensional space or trimming off the redundant features in the input space. Transformation of the input space to lower dimensional space is achieved by using a procedure like Principal Component Analysis (PCA) (Bishop, 1995). Engineering judgement and procedures like Automatic Relevance Determination (ARD) (MacKay, 1994; Neal, 1996; Neal, 1998) are used to prune the input space. Mdlazi et al. (2003) compared the performance of ARD to PCA focusing on the practical implementation issues of the two input-selection schemes using practical vibration examples.In this work, the interest is in the time domain representation of the gear vibration data and moreover, it is undesirable to lose any of the underlying dynamics within the input space, which could be the case when the input space is pruned.

The requirement for time domain representation of the gear vibration data and input space reduction results in the requirement of efficiently mapping the input space (rotation synchronised gear vibration signal) to the output space (time domain average of the vibration signal) using less data than would otherwise be used in the direct time-domain averaging procedure.

ANNs and SVMs are suitable in such applications because of their non-linear mapping and generalisation capabilities (Bishop, 1995; Gunn, 1998). In this paragraph the developed filtering technique is described. The main idea is to simulate direct TDA using artificial intelligence, in this case ANNs and SVMs. This approach has the potential of reducing the amount of vibration data that is required to calculate the TDA by direct averaging if the ANNs and SVMs can successfully map a fraction of the input pace to the output (TDA calculated by direct averaging of all the input data) as shown in Chapter 3. The performance of this filtering approach will therefore depend the non-linear mapping and generalisation capabilities of ANNs and SVMs. This filtering concept operates in two stages as shown in the block diagram in Figure 4.1.

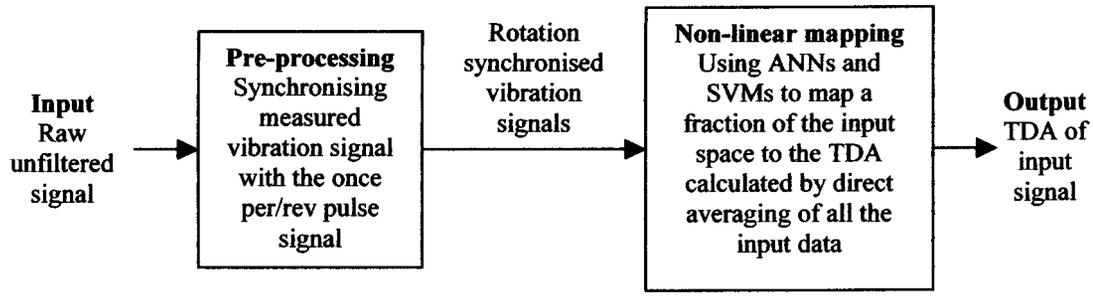


Figure 4.1 ANNs and SVMs base filtering concept.

More detail the different stages of this filtering technique are discussed later on in this chapter. Before much more is said about the data it is appropriate to describe the experimental set-up.

4.2.1 Experimental set-up

The data used in this study was obtained from the accelerated gear life test rig developed by Stander and Heyns (2002[a]) for their work on condition monitoring of gearboxes under fluctuating load conditions.

This experimental set-up consists of three Flender Himmel Motor helical gearboxes, driven by a 5.5 kW three-phase four pole WEG squirrel cage electric motor. A 5.5 kVA Mecc alte spa three-phase alternator was used for applying the load. The gear test rig was designed to conduct accelerated gear life tests on the Flender E20A gearbox under varying load conditions. Two additional Flender E60A gearboxes were incorporated in the design in order to increase the torque applied to the small Flender E20A gearbox. The rated load of the gears in the Flender E20A gearbox was 20 Nm. The Direct current (DC) fields of the alternator were powered by an external DC supply in order to control the load that was applied to the gears. A Hengstler R176T01 1024ED 4A20KF shaft encoder, which produced 1024 pulses, and 1 pulse per revolution in the form of an analog push-pull signal was used to measure the shaft speed. The reference point for synchronous averaging is measured as a single pulse from the shaft encoder. Acceleration measurements were taken in the vertical direction with a 5 V/g PCB integrated circuit piezoelectric industrial accelerometer and a Siglab model 20-42 signal analyser (Stander and Heyns, 2002[a]). The accelerated gear life test rig is shown in Figure 4.2.

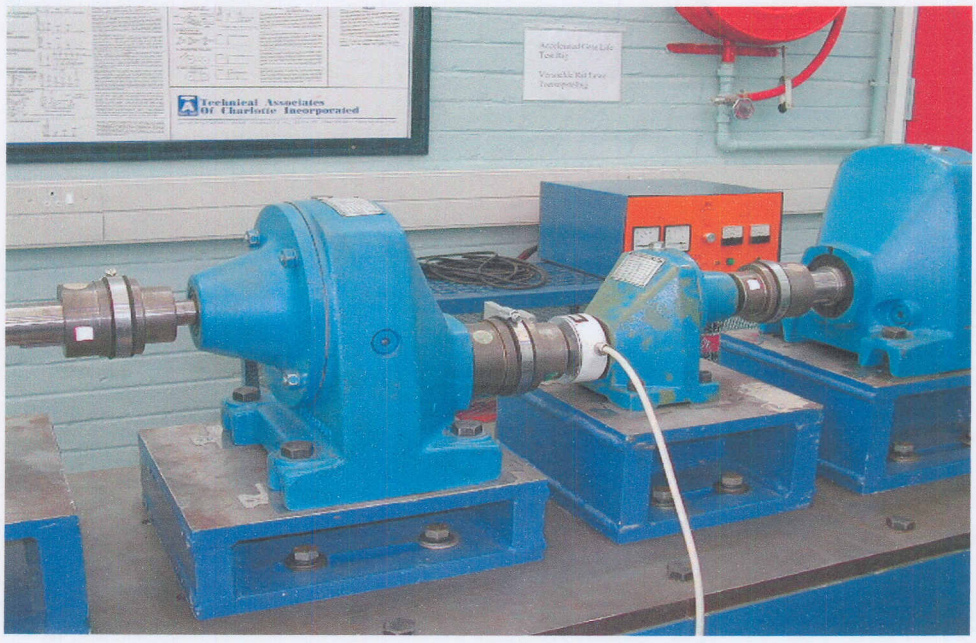


Figure 4.2 Accelerated gear life test rig.

The gears used in this experiment were manufactured in accordance with DIN3961, Quality 3. The gear specifications are given in Table 4.1 (Davel, 2003).

Table 4.1 Gear Specifications for gears used in accelerated gear life test rig.

Description	**Specification**
Helix angle at reference circle	30^{o}
Number of teeth (Pinion)	22
Number of teeth (Gear)	43
Nominal Module	1.250 mm
Base circle radius (Pinion)	14.64 mm
Base circle radius (Gear)	28.61 mm
Tip radius (Pinion)	17.55 mm
Tip radius (Gear)	0.331 mm

A typical signal obtained from the accelerometer is given in Figure 4.3 (a). This signal is synchronised with the pulse signal given in Figure 4.3 (b) in order to isolate the vibration that is produced by each gear rotation.

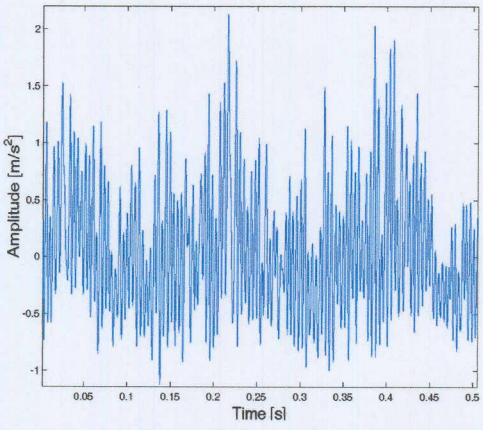


Figure 4.3 (a) Measured acceleration signal over 0.5 seconds.

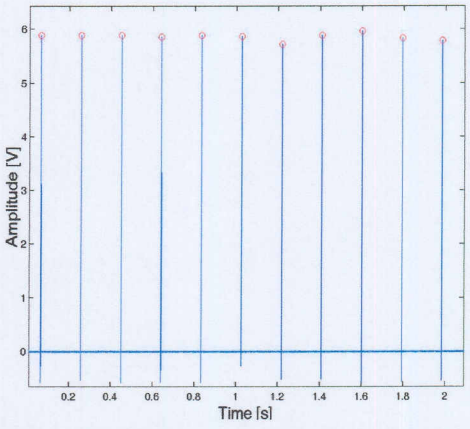


Figure 4.3 (b) One pulse per/rev shaft encoder signal for synchronising the gear vibration.

In order to assess the performance of the developed synchronous filter under both constant and varying load conditions, the measurements were also taken under different load conditions over the entire life of a gear. The load on the gearbox was applied by changing the current supplied to the alternator. The different loading conditions are given in Table 4.2.

Table 4.2 Experimental load conditions

Type of loading	Accelerated life test	Constant load	Sine function	Square function	Random function
Frequency [Hz]	0.0	0.0	0.5	0.3	2.0-5.12
Amplitude [V]	0.0	0.0	2.0	2.0	2.0
Offset [V]	5.0	3.0	3.0	3.0	3.0

4.2.2 Data processing

The acceleration data obtained from the gear test rig is passed through an eighth order low pass Butterworth filter with a cut-off frequency of 300 Hz. The acceleration signal is sampled at 51.2 kHz to ensure a true data representation. The filtered acceleration signal is synchronised with the pulse signal. The signals obtained after synchronising the low pass filtered acceleration signal with the pulse signal are not the same length because of inaccuracies with the pulse signal. The signals are therefore resampled so that they can have exactly the same period. In this work the signals were resampled to 8192 sample points per gear rotation. There were 165 gear rotations per test. Synchronising the measured acceleration signal with the shaft encoder signal and resampling each signal to 8192 resulted in an input space of (165×8192). For convenience, in this work an input space of (160×8192) was selected. We seek to predict the time domain average (synchronous average) of the gear vibration using only a fraction of this input space.

In Chapter 3 it was shown that 40 rotation synchronised gear vibration signals are suitable for predicting TDA of the gear vibration signals. This result implies that an input space of dimensions (40×8192) can be used to calculate the TDA instead of the input space of (160×8192) that would otherwise be used when the direct time domain averaging approach is used to calculate the TDA. Using 40 rotation synchronised gear vibration signals instead of 160 rotation synchronised gear vibration signals is a reduction of 75 percent in the input data requirement. This result also has some implications in terms of the training and validation data sets. As stated above each test produces 165 rotation synchronised gear vibration signals. If 40 of the rotation synchronised gear vibration signals are used to train the neural network then the other 125 rotation synchronised gear vibration signals can be grouped in sets of 40 and used as validation sets. For this section of the work measurements were taken under constant load at different stages of gear life.

4.2.3 Model 1

In this section a model that utilises ANNs and SVMs to predict the TDA of gear vibration is presented. A sensitivity study to assess the sensitivity of the proposed model to number of inputs, number of hidden units, number of sample points per revolution and percentage noise in the validation sets is also presented. The training and validation data sets for Model 1 are tabulated in Table 4.3.

Table 4.3. Model 1 training and validation data sets for tests conducted under constant loading.

Gear life stage	**Test 1** New gear	**Test 2** Running in gear	**Test 3** Midlife	**Test 4** Advanced damage
Training set	1×(160×8192)	0	0	0
Validation sets	3×(160×8192)	4×(160×8192)	4×(160×8192)	4×(160×8192)

The first model (Model 1) utilises a simple feedforward network structure as shown in Figure 4.4. This model attempts to map the input space (rotation synchronised gear vibration signals) to the target (time domain average of the rotation synchronised gear vibration signal) using feedforward network structure in a single step.

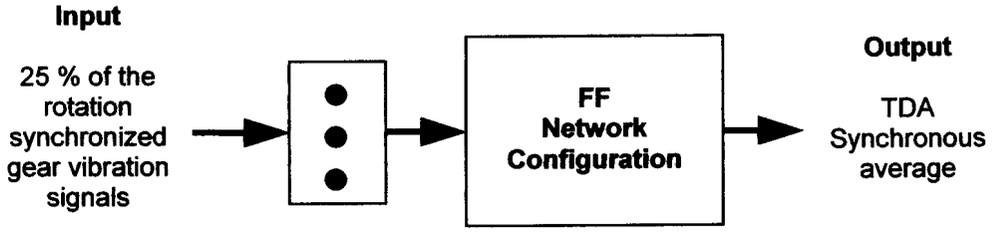


Figure 4.4 Model 1 with feedforward network structure.

Model 1 was investigated for all the formulations discussed in Chapter 3. The analysis of Model 1 with RBF and MLP was carried out in the following steps:

- The most suitable network architecture for mapping the input to the target is selected.

- The optimum number of inputs to the selected network architecture that correctly predicts the TDA of the gear vibration data using validation data sets are determined.
- The sensitivity of Model 1 with MLP or RBF networks to the number of sample points per revolution is assessed.
- A sensitivity study to assess the robustness of Model 1 with MLP or RBF networks with respect to noise is conducted.

4.2.3.1 Model 1 with MLP feedforward network

The first network formulation that was investigated is the MLP network. For convenience, the root mean square error given in Equation (4.1) was used to assess the performance of the neural network for the proposed model.

$$RMSE_k = \sqrt{\frac{1}{N_v}\sum_{i}^{N_v}\left(t_k^{(i)} - y_k^{(i)}\right)^2} \tag{4.1}$$

N_v is the number of validation data, $t_k^{(i)}$ is the target and $y_k^{(i)}$ is the network output. The optimum number of nodes and inputs were determined in order to correctly approximate the time domain averaging process while avoiding overfitting and thus bad generalisation.

A suitable network architecture was selected by first randomly choosing a network structure. The randomly selected structure was optimised sequentially by changing the network parameters while monitoring the *RMSE*, until a satisfactory time domain average prediction was obtained. The results obtained for analysis of the MLP using 20 unseen validation sets of dimensions (40×8192) are presented below. Figure 4.5 is a plot of *RMSE* against the number of hidden units.

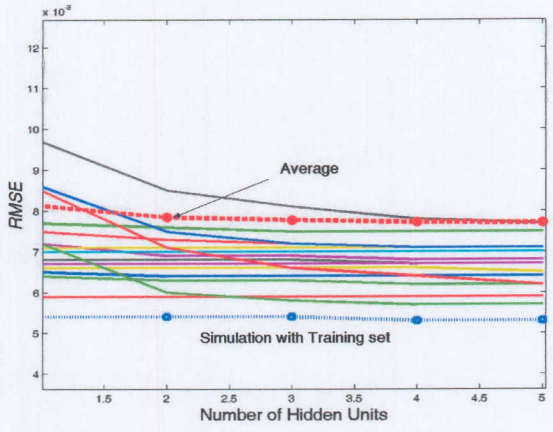


Figure 4.5 *RMSE* vs. Number of hidden units for 20 validation sets.

This plot shows that simulation with the training set is insensitive to the number of hidden units. When simulating with the validation sets there is general decrease in the *RMSE*, then it stabilises at four hidden units. This is because increasing the number of hidden units increases the flexibility of the network, thus increasing its capability to map the input to the output. This plot also shows the average of all the simulations with validation sets. The average shows insensitivity to the number of hidden units because most of the validation sets are fairly insensitive to the number of inputs. The average is somehow deceptive because some of the validation sets are sensitive to the number of inputs. The average can, however, still be useful for comparison purposes because the majority of the validation sets are insensitive to the number of hidden units. In this study an MLP network with 5 hidden units was selected. Figure 4.6 shows the *RMSE* plotted against the number of inputs for Model 1 simulated with 20 unseen validation sets.

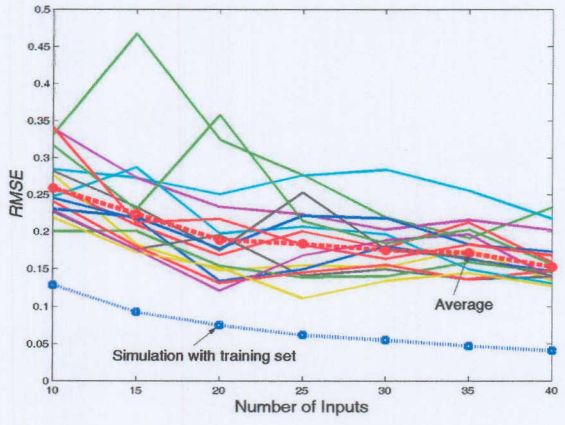


Figure 4.6 *RMSE* vs. Number of inputs for simulation with 20 validation sets.

Figure 4.6 shows the *RMSE* plotted against the number of inputs. In Chapter 3 it was established that 40 inputs were the optimal number of inputs for mapping the rotation synchronised gear vibration signals to the TDA, therefore, 40 inputs were selected as the maximum number of inputs in this analysis. In Figure 4.6 it is observed that the *RMSE* decreases as the number of inputs is increased. This is expected because the network is exposed to more of the underlying system dynamics, and therefore, trains and predicts more efficiently than would otherwise be the case with less inputs. The average of simulations with the validation sets and the simulation with the training set are also plotted. These plots also confirm the fact the as the network is exposed to more inputs, its prediction capabilities are enhanced. One of the objectives of this work is to decrease the number of inputs that are required to calculate the TDA, therefore, the analyst needs to make a compromise between the network accuracy and the number of inputs that are used. Figure 4.7 shows the *RMSE* plotted against the number of sample points per gear rotation.

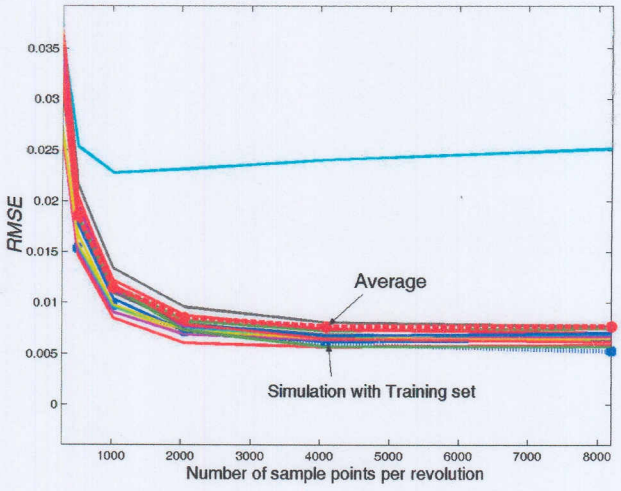


Figure 4.7 *RMSE* vs. number of sample points per revolution for 20 validation sets.

To assess the sensitivity to the number sample points per rotation synchronised gear vibration signal, the 20 validation sets with 40 inputs were resampled from 8192 sample points down to 4096, 2048, 1024, 512 and 256 respectively. Figure 4.7 shows the *RMSE* plotted against the number of sample points per revolution. The *RMSE* decreases until 2048 points per revolution, and thereafter remains constant. For the MLP simulation 1024 points per revolution were used to reduce computational load though 2048 sample points per revolution would have been more suitable. In Figure 4.7 it is

observed that there is a plot that is an outlier. This plot corresponds to the validation set that corresponds to the damaged gear. This is because vibration produced by the damaged gear is very different to that produced throughout the life of the gear, and the selected network did not generalise well to this condition. Figure 4.8 shows the *RMSE* plotted against percentage noise in the validation set.

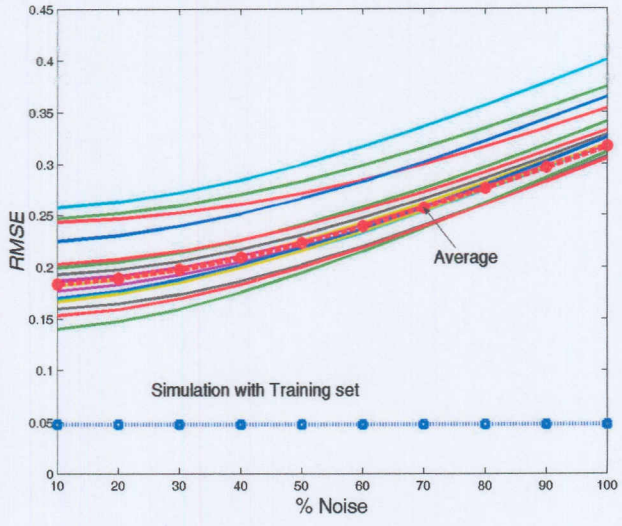


Figure 4.8 *RMSE* vs. percentage Noise in validation set for simulation with 20 validation sets.

Figure 4.8 shows the *RMSE* plotted against the percentage of random noise that is superimposed on the validation sets. The random noise is chosen from a normal distribution with a zero mean and variance of one. There is a direct relationship between the noise level and the *RMSE*. From this analysis Model 1 that has a MLP feedforward network was found to be fairly tolerant to noise.

4.2.3.2 Model 1 with RBF feedforward network

The second network formulation that was investigated for use in Model 1 is the RBF network. The procedure that was followed in the analysis of the MLP network was used for the RBF network. This section presents the results from the analysis of Model 1 with the RBF network. The sensitivity of Model 1 to number of inputs, number of hidden units, number of sample points per revolution and percentage noise superimposed on the validation sets is also presented. Figure 4.9 shows the *RMSE* plotted against number of hidden units (number of basis functions).

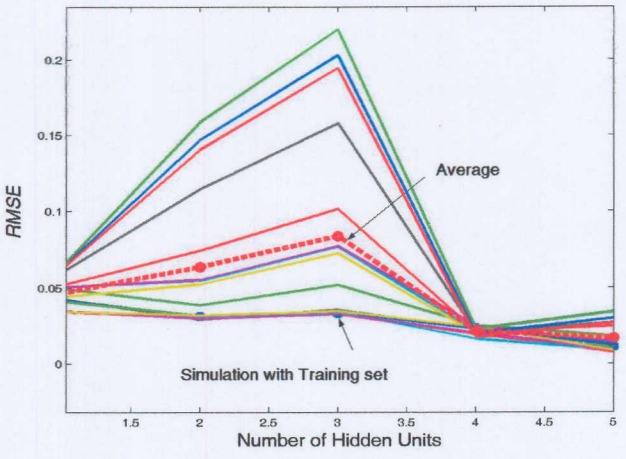


Figure 4.9 *RMSE* vs. number of hidden units for 20 validation sets.

Figure 4.9 shows the *RMSE* plotted against the number of hidden units (Number of basis functions). It is observed from this plot that the RBF is very sensitive to the number of basis functions. From Figure 4.9 it is observed that Model 1 with RBF network results in best generalisation at 4 basis functions, therefore 4 basis functions were selected for this study. It is possible that there is another optimum when more than 5 basis functions are used. To keep the computational load at a minimum it was decided to settle for 4 basis functions. Figure 4.10 shows the *RMSE* plotted against the number of inputs for Model 1 simulated with 20 unseen validation sets.

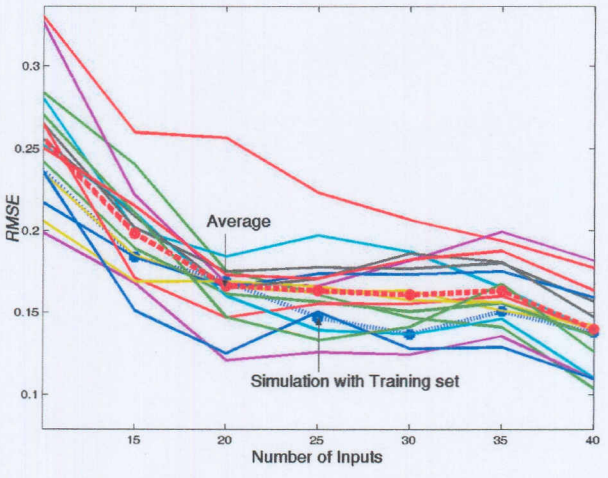


Figure 4.10 *RMSE* vs. number of inputs for simulation with 20 validation sets.

Figure 4.10 shows the *RMSE* plotted against the number of inputs. Again as observed for MLP network, the number *RMSE* decreases as the number of inputs is increased. For

RBF, there is a definite minimum for all the validation sets at 40 inputs. For this work 40 inputs were selected as the optimum number of inputs for Model 1 with RBF. Figure 4.11 shows the *RMSE* plotted against the number of sample points per gear rotation.

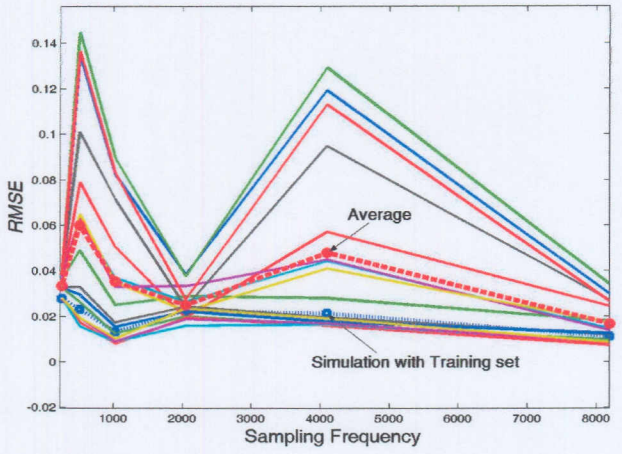


Figure 4.11 *RMSE* vs. number of sample points per revolution for 20 validation sets.

From this plot it is observed that the optimum number of sample points per revolution is 2048 but for computational efficiency 1024 points were selected. In Figure 4.12 the *RMSE* is plotted against the percentage noise in the validation set.

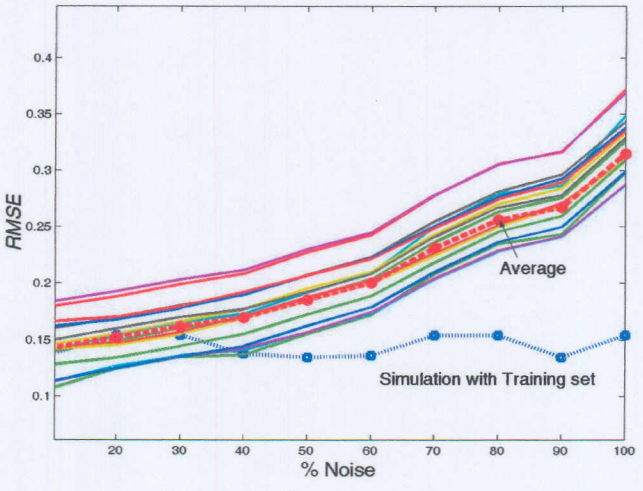


Figure 4.12 *RMSE* vs. percentage Noise in validation set for simulation with 20 validation sets.

Figure 4.12 is a plot of the *RMSE* against the percentage of random noise that is superimposed on the validation sets. There is again a direct relationship between the percentage noise and the *RMSE*. As the percentage noise increases so does the *RMSE*.

4.2.3.3 Model 1 with SVMs

This section presents the results from the analysis of Model 1 with SVMs. The sensitivity of Model 1 to number of inputs, number of hidden units, number of sample points per revolution and percentage noise superimposed on the validation sets is also presented.

The analysis of Model 1 with SVMs was carried out in the following steps:

- The most suitable Kernel function for mapping the input space to the target was selected.
- The optimum number of inputs for correctly predicting the TDA of the rotation synchronised gear vibration signals was determined.
- The sensitivity of Model 1 with SVMs to percentage noise in the validation sets was assessed.

Figure 4.13 shows the plot of the performance of different Kernel functions that were investigated in this study for the non-linear SVM regression task. In Figure 4.13 the *RMSE* produced by each of the Kernel function on an unseen validation set is plotted against the order of the Kernel function. From this plot, it is observed that the Exponential Radial Basis Function (ERBF) Kernel function outperforms the other Kernel functions. It is also observed that the ERBF Kernel function is insensitive to the order of the Kernel function. In this study the ERBF Kernel function with an order of 10 was therefore selected.

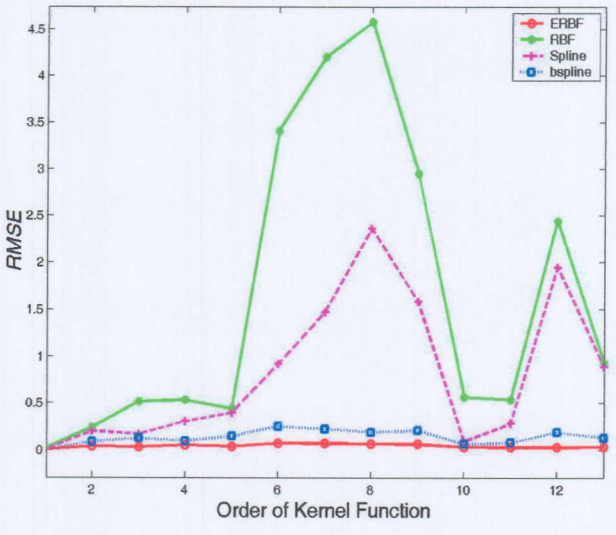


Figure 4.13 RMSE vs. Order of Kernel function an unseen validation sets.

Figure 4.14 shows the *RMSE* plotted against the number of inputs for Model 1 simulated with 20 unseen validation sets. It is observed from Figure 4.14 that simulation with the training set has a constant prediction error irrespective of the number of inputs. This is an indication of the robustness of the SVM algorithm. Simulation with validation sets indicates that there is an inverse proportionality between the number of inputs and the prediction error. This is because the SVM is exposed to more of the system dynamics as the number of inputs is increased; therefore, it trains more effectively.

SVMs are computationally expensive, therefore, in this work 256 points per rotation synchronised gear vibration signal were selected for all SVM analyses.

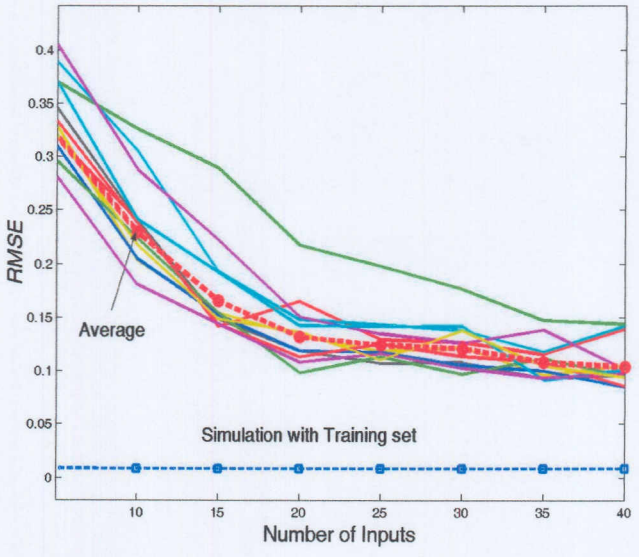


Figure 4.14 *RMSE* vs. number of inputs for 20 validation sets.

Figure 4.15 shows the *RMSE* plotted against percentage noise. A direct relationship between the *RMSE* and the percentage noise is observed. This is expected because the introduction of noise to the validation sets increases the degree of nonlinearity between the input and output space.

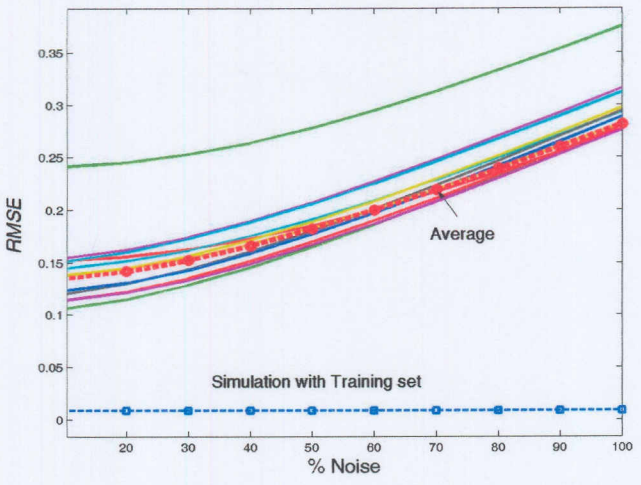


Figure 4.15 *RMSE* vs. percentage noise in validation set for simulation with 20 validation sets.

4.2.4 Model 2

In this section a second model (Model 2) that utilises ANNs and SVMs to predict TDA of gear vibration is presented. Model 2 estimates the TDA of the input space in small sequential steps, analogous to taking a running average of the input space. This model consists of a number of small feedforward networks similar to those in Model 1 but instead of the networks being used to predict the TDA of the whole input space in one step, the small feedforward networks are used to first sequentially predict the average of subsections of the input space (instantaneous averages). The output of the first set of feedforward networks are used as inputs into a second feedforward network that predicts the time domain average of the whole input space. All the feedforward networks are trained off-line to reduce computational time. In this model all the data that have already been used can be discarded immediately. This means that one does not need to store large amounts of data in the data collection system. The training and validation data sets for Model 2 are tabulated in Table 4.4.

Table 4.4. Model 2 training and validation data sets for tests conducted under constant loading.

Gear life stage	**Test 1** New gear	**Test 2** Running in gear	**Test 3** Midlife	**Test 4** Advanced damage
Training set	1×(10×8192)	0	0	0
Validation sets	15×(10×8192)	16×(10×8192)	16×(10×8192)	16×(10×8192)

In this section a sensitivity study to assess the sensitivity of the proposed model to number of inputs, number of hidden units, number of sample points per revolution and percentage noise in the validation sets is presented. A schematic diagram of Model 2 is shown in Figure 4.16.

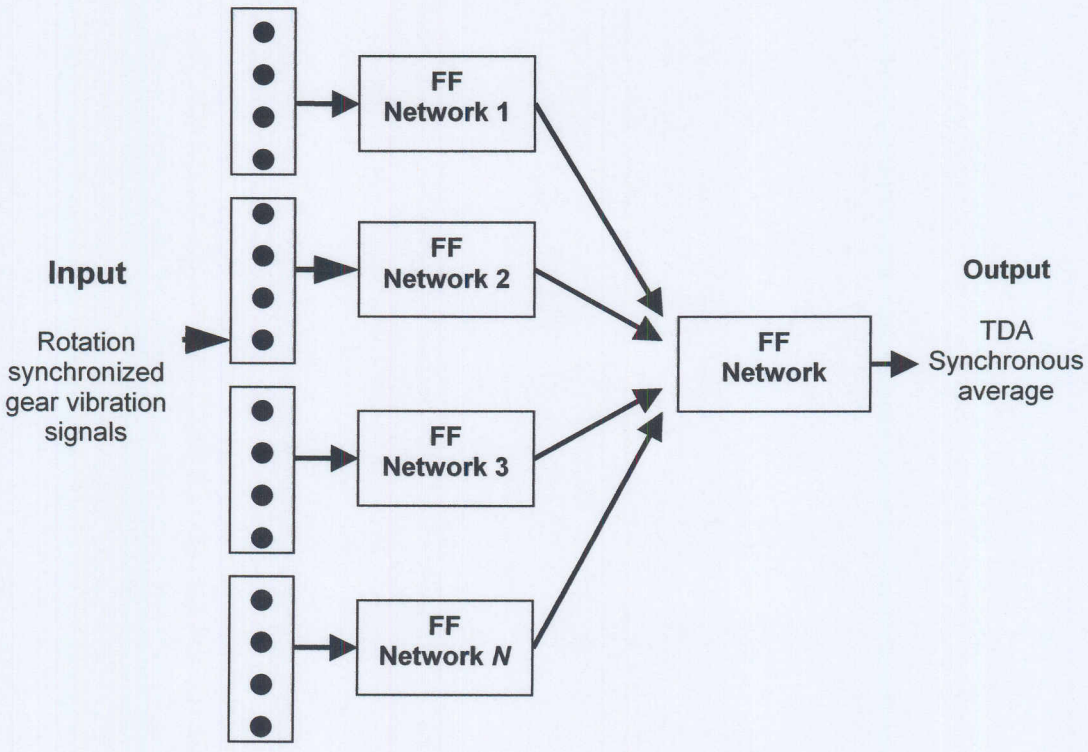


Figure 4.16 Model 2

Model 2 was investigated for all the formulations discussed in Chapter 3. The analysis of Model 2 with RBF and MLP was carried out in the following steps:

- The most suitable network architecture for mapping a small section of the input to its average (target) is selected.
- The sensitivity of Model 2 with MLP or RBF networks to noise is assessed.
- In Section 4.2.3.1 and Section 4.2.3.1 the sensitivity of Model 1 with MLP and RBF networks to the number of sample points per rotation synchronised gear vibration was assessed. It was concluded the 1024 points per revolution are suitable, therefore 1024 sample points per rotation synchronised gear vibration signal are selected for Model 2.

4.2.4.1 Model 2 with MLP feedforward networks

This section presents the results from the analysis of Model 2 with MLP feedforward networks. Figure 4.17 shows the simulation results of Model 2 with MLP network simulated using an unseen validation set. The simulation result is superimposed on TDA calculated using the direct time domain averaging approach.

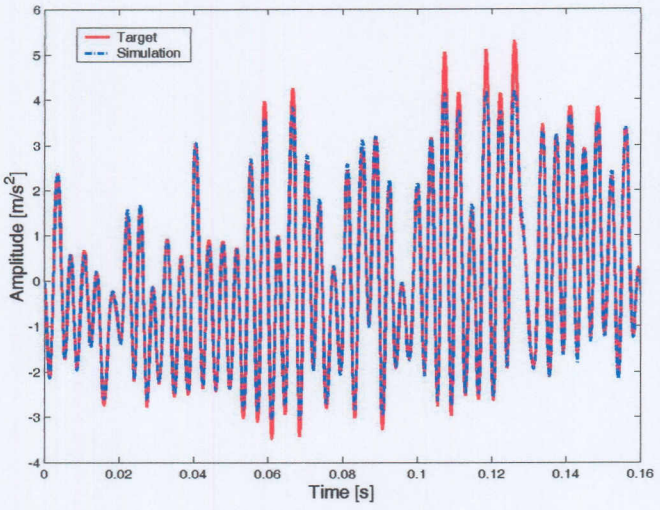


Figure 4.17 Model 2 with MLP network simulation result for an unseen validation set superimposed on TDA after 160 gear rotations.

This plot shows that Model 2 with MLP feedforward networks is very suitable for predicting the time domain average of the gear vibration. This is because Model 2 uses the whole input space to predict the TDA as opposed to Model 1 that uses only a section of the input space. Figure 4.18 shows the *RMSE* plotted against percentage noise in the training and validation sets. The random noise content is again chosen from a normal distribution with a zero mean and variance of one. For the simulation with the training set it is observed that the model is fairly insensitive to noise. This is, however, not the case when simulating with validation sets. There is a direct relationship between the *RMSE* and the percentage noise superimposed in the data. This is because addition of noise to the training and validation sets increases the degree of non-linearity between the input space and the output.

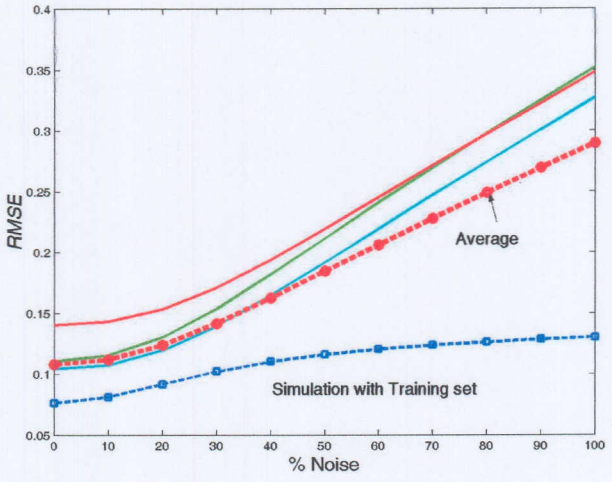


Figure 4.18 *RMSE* vs. percentage noise in training and validation sets for 3 unseen validation sets.

4.2.4.2 Model 2 with RBF feedforward networks

This section presents the results from the analysis of Model 2 with RBF feedforward network. In Figure 4.19 the simulation results of Model 2 with RBF network simulated using an unseen validation set is superimposed on the TDA calculated using direct averaging. The predicted TDA in Figure 4.19 is almost an exact fit. This is because Model 2 uses the whole input space to predict the TDA; therefore the network is exposed to all the transient effects in the data.

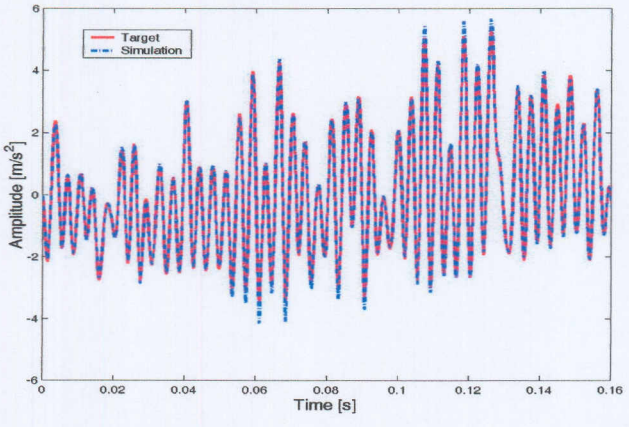


Figure 4.19 Model 2 with RBF network simulation result for an unseen validation set superimposed on TDA after 160 gear rotations

Figure 4.20 shows the *RMSE* from 3 validation sets plotted against percentage noise in the training and the validation sets. Both the training and the validation sets exhibit a linear relationship between the *RMSE* and the percentage noise superimposed in the

data. This is again because addition of noise to the training and validation sets increases the degree of non-linearity between the input space and the output.

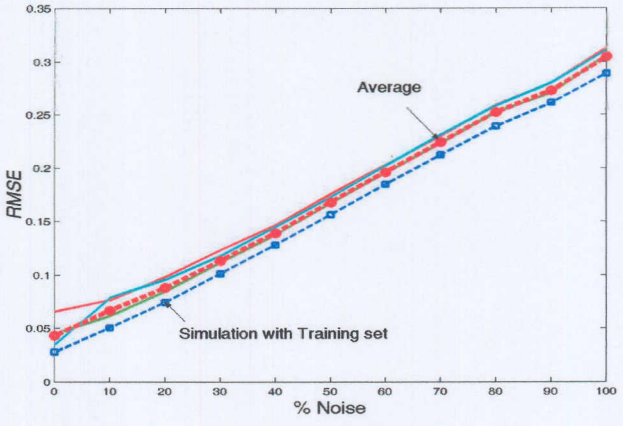


Figure 4.20 *RMSE* vs. percentage noise in training and validation sets for 3 unseen validation sets.

4.2.3.3 Model 2 with SVMs

This section presents the results from the analysis of Model 2 with SVMs in place of feedforward neural networks. Figure 4.21 shows that the SVM is also suitable for use in this modal with its only drawback being computational inefficiency. The good performance of Model 2 is because it uses the whole input space to predict the TDA; therefore the SVM is exposed to all the transient effects in the data. To maintain the computation time at a minimum, 256 samples points per rotation synchronised gear vibration signal were selected.

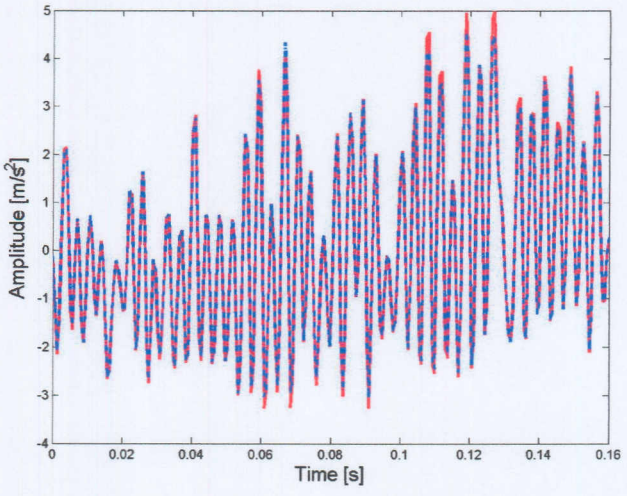


Figure 4.21 Model 2 with SVM simulation results with an unseen validation set superimposed on TDA after 160 gear rotations

Figure 4.22 shows the *RMSE* from 3 validation sets plotted against percentage noise in the training and the validation sets. With SVM the simulation with training set shows that the support vector machine is fairly insensitive to noise. This is not the case when simulating with the validation sets. Simulations with the validation sets indicate a direct relationship between the *RMSE* and the percentage noise in the data. The observed direct relationship is due to the fact that the addition of noise to the training and validation sets increases the degree of non-linearity between the input space and the output.

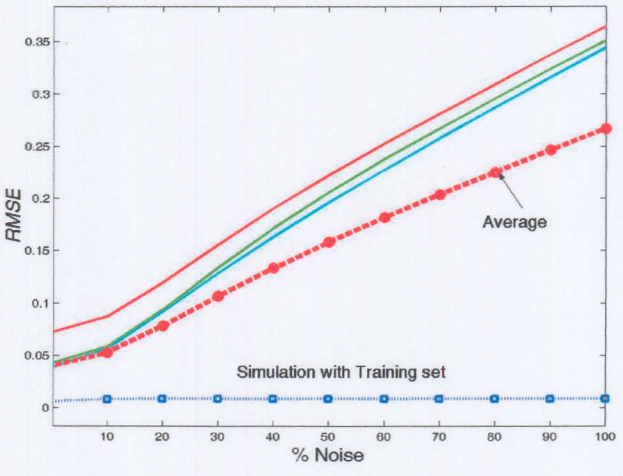


Figure 4.22 *RMSE* vs. percentage Noise in training and validation sets for 3 unseen validation sets.

4.2.5 Discussion

In this section a comparison between the different mathematical formulations is presented. The comparison is based on the average *RMSE* of simulation with a population of 20 validation sets (validation examples) for Model 1 and 3 validation sets (validation examples) for Model 2 an computation time.

4.2.5.1 Model 1

Figure 4.23 to Figure 4.26 show the results obtained for different formulations as functions of some network parameters. Figure 4.23 shows the *RMSE* plotted against the number of hidden units. From this plot it is observed that the MLP network performs better than the RBF network. The RBF network is very sensitive to the number of hidden units while the MLP is fairly insensitive. The insensitivity of MLP to the number of hidden units is desirable because it implies that smaller and more computationally

of hidden units is desirable because it implies that smaller and more computationally efficient networks can be used to calculate the TDA. Figure 4.24 shows the *RMSE* plotted against the number of inputs for Model 1 with MLP, RBF networks and SVM respectively. From this plot is observed that Model 1 with SVMs results in the best performance especially at higher number of inputs. The superior performance is because of the structural risk minimisation used in SVMs is superior in generalisation to the empirical risk minimisation used in neural network (Gunn et al., 1998). The performance of Model 1 with RBF and MLP networks are comparable although the RBF performs slightly better than MLP.

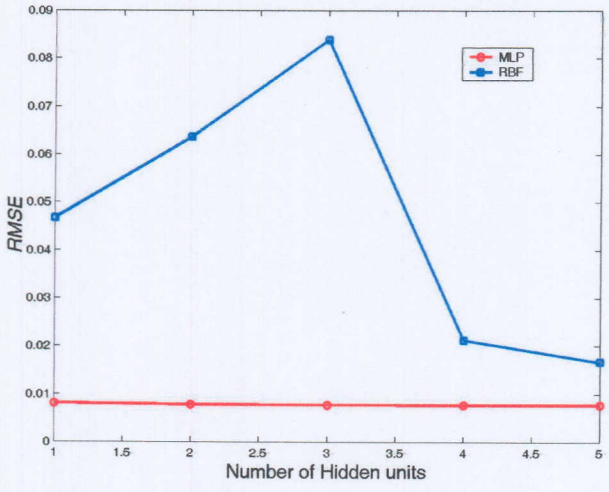


Figure 4.23 *RMSE* vs. number of hidden units for Model 1 with MLP and RBF networks.

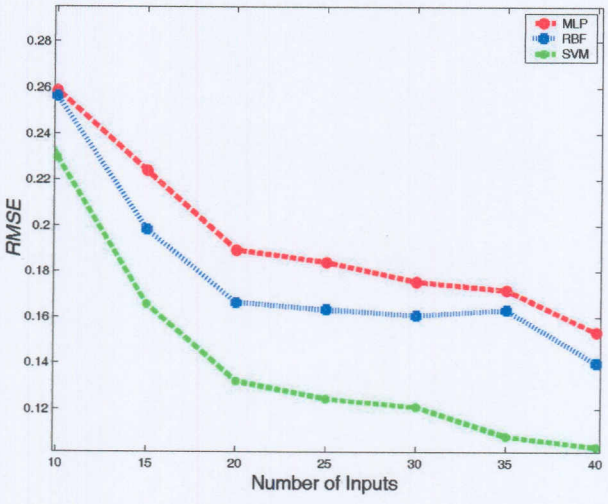


Figure 4.24 *RMSE* vs. number of inputs for Model 1 with MLP, RBF and SVMs.

In Figure 4.24 the reduction in *RMSE* as the number of inputs is increased is because the formulations are exposed to more of the underlying system dynamics as the number of inputs is increased, therefore they train more efficiently. Figure 4.25 shows *RMSE* plotted against the number of sample points per revolution. This plot shows that Model 1 with MLP network performs better than Model 1 with RBF network, which is very sensitive to the number of sample points per revolution. Figure 4.26 shows the performance of the three formulations as a function of percentage noise in the training and validation sets. This plot shows that the SVM again performs better than both MLP and RBF although the performances are fairly comparable. This is due to the superior performance of the structural risk minimisation used in SVMs (Vapnik, 1995; Gunn, 1998).

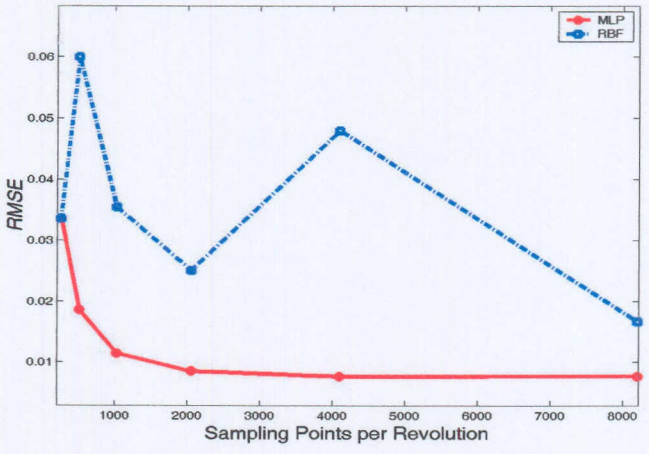


Figure 4.25 *RMSE* vs. number of sample points per revolution for Model 1 with MLP and RBF networks.

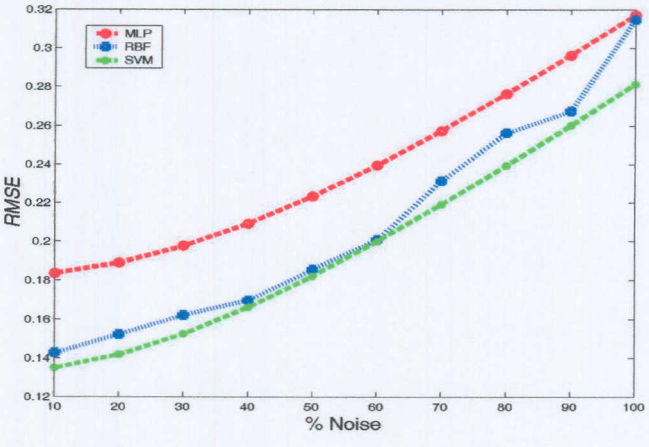


Figure 4.26 *RMSE* vs. percentage noise in training and validation sets for Model 1 with MLP, RBF and SVMs.

4.2.5.2 Model 2

Figure 4.27 and Figure 4.28 show the performance of Model 2 with the three formulations. Figure 4.27 shows the *RMSE* produced by 3 different validation sets. The SVM performs very well when simulating with the training set but performs poorly for validation sets. MLP and RBF both perform better than the SVMs for the validation sets. The poorer performance of Model 2 with SVM as compared to Model 2 with MLP and RBF networks is due to the fact that MLP and RBF are more suited to Model 2.

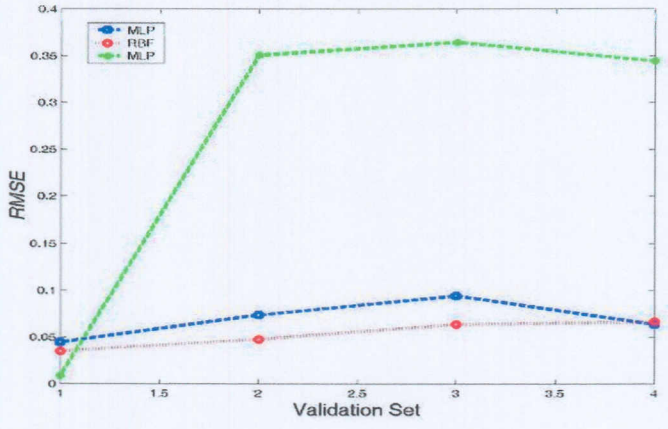


Figure 4.27 *RMSE* vs. validation set for 3 validation sets

Figure 4.28 shows the performance Model 2 with MLP, RBF and SVMs as a function of the percentage noise in the training and validation sets. It is observed that the performance of these formulations is the same. The observed direct relationship between the percentage noise in the training and validation sets and the *RMSE* is because the addition of noise increases the degree of non-linearity between the input space and the output.

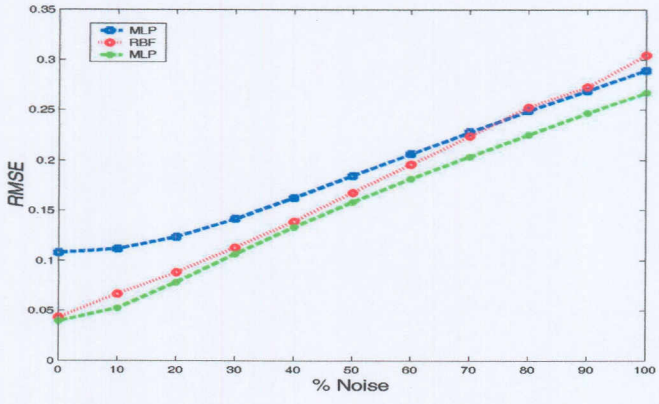


Figure 4.28 *RMSE* vs. percentage noise in training and validation sets for Model 2 with MLP, RBF and SVMs.

4.2.5.3 Computation time

To put the proposed methods in perspective, a comparison of computation times for the existing time domain averaging method and the proposed models is done. A Pentium 4 computer with a 1.60 GHz processor was used. The computation times are presented in Table 4.5.

It is observed that the required pre-processing time for Model 1 is less than the required pre-processing time for the TDA calculated by direct averaging. This is because Model 1 uses 25 percent of the vibration data as opposed to the original TDA process, which uses the all the vibration data. The required pre-processing time for Model 2 is equal to the pre-processing time for the TDA calculated by direct averaging. This is because both models use the same amount of vibration data. When Model 1 and Model 2 are used, RBF and MLP give the best performance in terms of simulating time and SVMs gives the poorest performance. The models are trained off-line therefore the training time does not influence the simulating time in a real time applications. When the models are used with MLP and RBF networks, they perform much better than original TDA calculated by direct averaging in terms of simulation time. It is however, observed that when the models are used with SVMs, the performance is much poorer than the performance of the TDA computed by direct averaging.

The poor performance in SVMs is because the training problem is a quadratic optimisation problem with 2N variables, where N is the number of data training points. Thus the more the data that is used when train, the longer it takes. This is much slower than the MLP and RBF neural networks in which only the weights and biases or the basis centres are obtained by minimising error functions.

Table 4.5 Computation time

	TDA	**MLP**	**RBF**	**SVM**
Model 1 Pre-processing time [s]	1.011	0.703	0.703	0.703
Model 1 Training time [s]	-	22.24	2.219	497.0
Model 1 simulating time [s]	0.75	0.016	0.047	5.500
Model 2 Pre-processing time [s]	1.011	1.011	1.011	1.011
Model 2 Training time [s]	-	1.14	1.015	963.8
Model 2 simulating time[s]	-	0.08	0.078	83.76

4.3 Conclusion

In this chapter the use of MLP, RBF neural networks and SVMs in the development of a time domain averaging filter for gear vibration was investigated. It was shown that the amount of input vibration data required to calculate the TDA can be effectively reduced using ANNs and SVMs to predict the TDA of a gear vibration signal.

Two different filter models are considered. The first model (Model 1) uses a feedforward ANNs or SVMs to map input space (rotations synchronised gear vibration signals) to the target (time domain average after 160 shaft rotations). Using Model 1 a data reduction of 75 percent was achieved with all the formulations because Model 1 predicted the TDA using 40 of the 160 rotation synchronised gear vibration signals. Any of the three formulations can be used in this model because their performances are comparable although the SVMs may seem to be more attractive. Its attractiveness is reduced by the fact that it is more computationally expensive than MLP and RBF, therefore, the analyst will need to be cautious when SVMs are implemented in an online system that is required to retrain regularly. On the other hand the MLP and RBF networks are quick and easy to train, therefore they are suitable for implementation in an online system even when required to retrain online.

The second model (Model 2) operates in two stages. In the first stage it uses 10 inputs (10 rotations synchronised gear vibration signals) to predict the instantaneous time domain average of the gear vibration. The input data to the first stage is deleted from the memory of the data acquisition system after it has been used. The output of the first stage is used as input to a second feedfoward network to predict the TDA. This means that the largest number of rotation synchronised gear vibration signals that will be stored in the data acquisition system is 26 inputs. This is a reduction of 83 percent in the amount of data that needs to be stored in the data acquisition system. It must be noted however that, with Model 2, the entire data set is used although it use sequentially. Model 2 was found to be very effective at predicting the time domain average for all three formulations. In Model 2 the MLP and RBF perform better than SVMs.

Chapter 5

Testing the synchronous filter on experimental data

5.1 Introduction

In this chapter the synchronous filter for time domain averaging of gear vibration data that was developed in Chapter 4 is implemented on a new data set from the accelerated gear life test rig. The data set was measured at two-hour intervals throughout the gear life. Taking measurements over the entire gear life serves the purpose of demonstrating the suitability of the developed filter in predicting the time domain average over the entire gear life. Secondly, we want to evaluate the suitability of the developed synchronous filter for time domain averaging for use in cases where the applied load is not constant as would be the case in a typical industrial application. In this a brief background on the data is presented. This is followed by simulation using the developed models for MLP, RBF and SVM, respectively. The results are compared to those that are obtained using direct time domain averaging approach, focusing on the practical implications. The synchronous filter is also tested on data from a test conducted under varying load conditions.

5.2 Data representation

The gear was expected to have a life of 30 hours; therefore vibration measurements were taken in two-hour intervals until failure occurred so as to properly monitor the progression of gear failure. The data was sampled at a frequency of 51,2 kHz to get a full representation of the frequencies of interest, in our case, the gear mesh frequency (*GMF*) and its side bands (*SB*). The *GMF* is defined by

$$GMF = S_f \times N_T, \tag{5.1}$$

where S_f is the shaft frequency and N_T is the number of gear teeth. The side bands (*SB*) occur at a frequency F_r defined by

$$F_r = GMF - kS_f, \tag{5.2}$$

where k is an integer and S_f is the shaft frequency. For the accelerated gear life test rig the *GMF, SB* and operating properties are given in Table 5.1.

Table 5.1 Operating properties for test data (Davel, 2003)

Rotational Speed [revs/min]	***GMF* [Hz]**	***SB.1* [Hz]**	***SB.2* [Hz]**
311	223.0	217.2	228.2

From Table 5.1 it is observed that the highest frequency of interest is 228.2 Hz therefore the measured acceleration was low-pass filtered at 300 Hz. In a gearbox where the applied load is constant, the amplitude of the side bands of the meshing frequency (*SB*) and gear mesh frequency (*GMF*) in the frequency spectrum are expected to increase as the vibration increases. This observation supplies a means of representing the progression of gear life using gear vibration. In this work the *SB* is normalised with the *GMF* and then plotted over the gear life. Figure 5.1 shows the FFT spectrum of the measured vibration signals from one of the test signals. This plot clearly shows the *GMF* and *SB*.

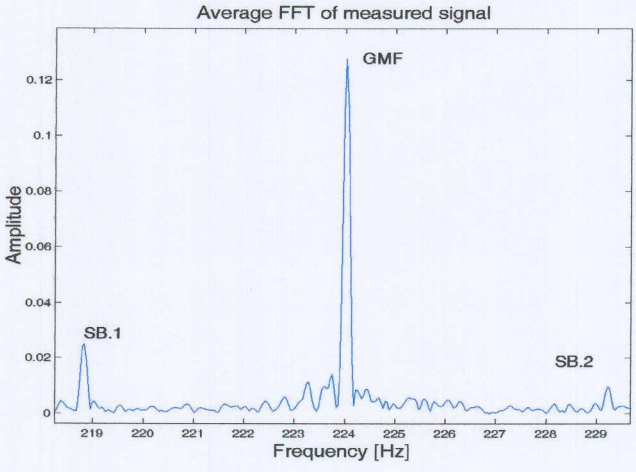


Figure 5.1 FFT plot measured acceleration signal after 9 hours in operation.

Figure 5.2 below shows the representation of the gear vibration over the entire gear life using normalised *SB.2*. This plot shows the normalised SB and a fourth order

polynomial fitted onto the normalised *SB* data. The fitted curve closely resembles the well known bath- tub curve (Norton, 1989) that models the life of most mechanical systems. The first 5 hours in operation show the running in stage of the gear life. The gear vibration level stabilises from 5 hours until 25 hours, which can be considered as constant gear wear stage. From 25 hours the gear vibration increases until the gear fails after 33 hours. The final stage can be considered as the wear out stage of the gear life.

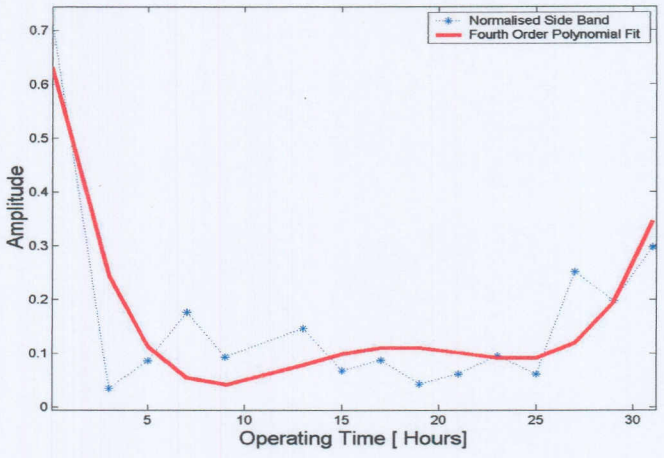


Figure 5.2 Testing data representation

In this chapter data sets from different stages of the gear life are used.

5.3 Model 1

This paragraph presents the results obtained using Model 1 for test data obtained from different stages of the gear life for tests conducted under constant load conditions as presented by Table 4.2. Model 1 with MLP, RBF and SVM formulations is tested and the results obtained for each of the formulations are compared to the TDA calculated using the direct time domain averaging approach after 160 gear rotations. For Model 1, 40 gear rotations are considered as the optimum number of inputs as discussed in Chapter 4. This is a reduction of 75 percent of the data that would be used when the direct averaging approach is used to calculate the TDA.

5.3.1 Model 1 with MLP feedforward network

Figure 5.3 (a) to Figure 5.3 (c) shows the prediction from Model 1 with MLP feedforward networks superimposed on the time domain average (TDA) of the gear vibration signals obtained using direct averaging. The simulations were done using

unseen validation sets as presented in Table 4.3. From Figure 5.3 (a) to Figure 5.3 (c), it is observed that Model 1 with MLP feedforward networks can correctly predict with 40 gear rotations the TDA for 160 gear rotations over the entire gear life. The FFT of the TDA calculated by direct averaging and the FFT of the simulation results are exact fits throughout the life of the gear. This indicates that Model 1 with MLP feedforward networks retains the diagnostic enhancing capabilities of TDA.

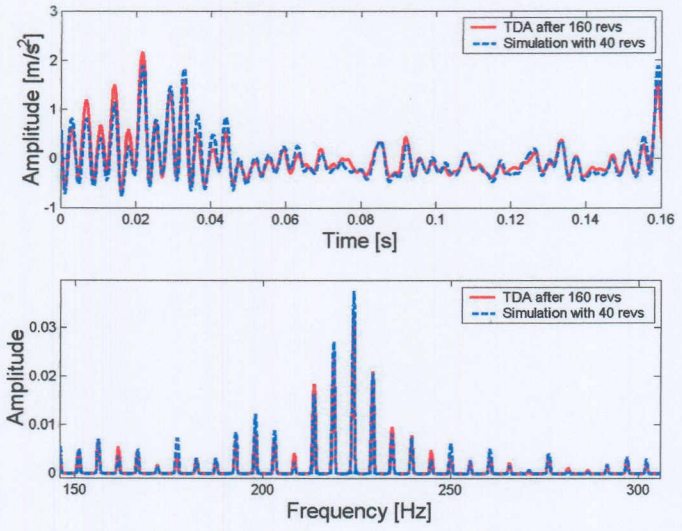


Figure 5.3 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations. The measurements were taken during the running in stage of the gear.

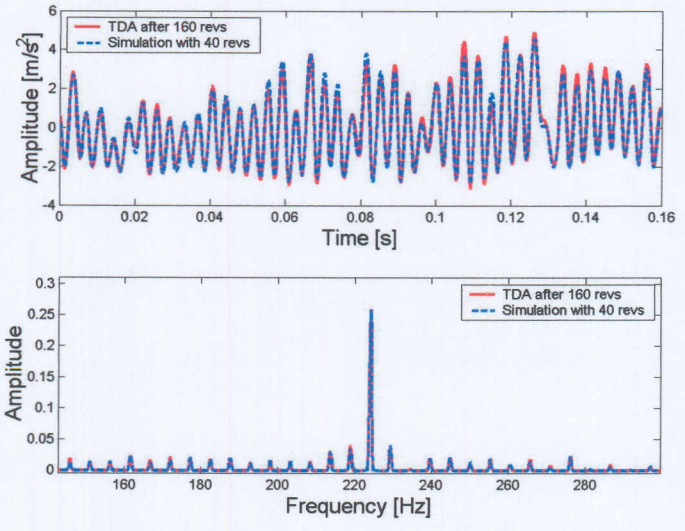


Figure 5.3 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations. The measurements were taken during the constant wear in stage of the gear.

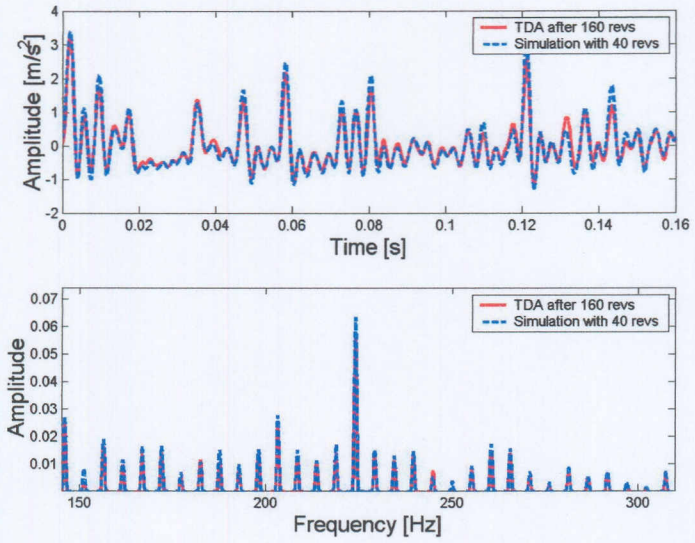


Figure 5.3 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations. The measurements were taken in the wear out stage of the gear.

5.3.2 Model 1 with RBF Feedforward network

Figure 5.4 (a) to Figure 5.4 (c) show the prediction of Model 1 with RBF feedforward neural networks. The simulation results are superimposed on the TDA of the gear vibration signals calculated using the direct averaging. The validation set had 40 inputs.

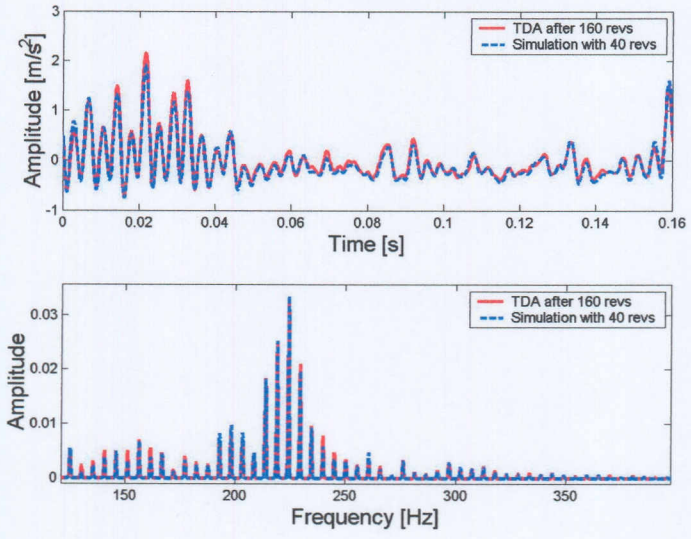


Figure 5.4 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations. The measurements were taken during the running in stage of the gear.

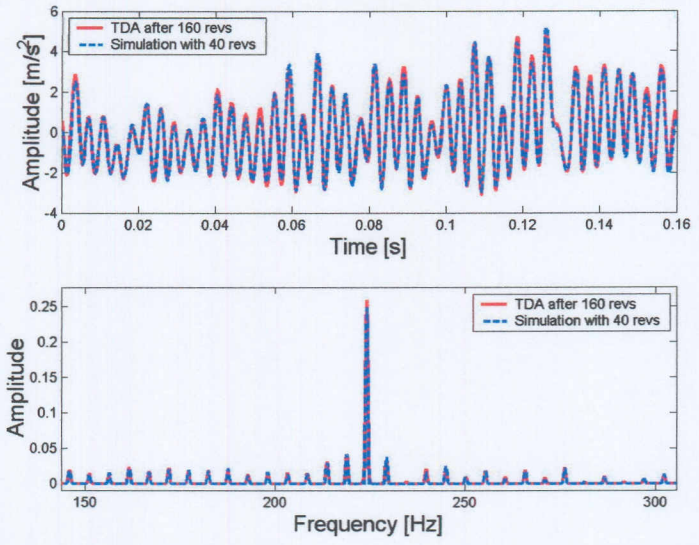


Figure 5.4 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA after 160 rotations. The measurements were taken during the constant wear stage of the gear.

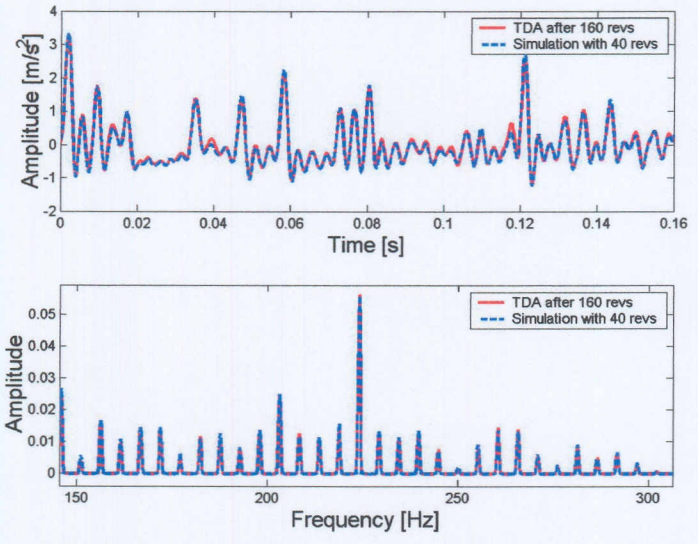


Figure 5.4 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA after 160 rotations. The measurements were taken during the wear out stage of the gear.

The above plots show in both frequency domain and time domain representation, that Model 1 with a RBF feedforward network with 40 gear rotations can correctly predict the TDA for 160 gear rotations over the entire life of the gear. This is because the selected RBF network can effectively map the input to the output space. The FFT of the TDA calculated by direct averaging and the FFT of the simulation results are exact fits

throughout the life of the gear. This indicates that Model 1 with RBF feedforward networks retains the diagnostic enhancing capabilities of TDA.

5.3.3 Model 1 with SVMs

Figure 5.5 (a) to Figure 5.5 (c) show the prediction obtained using Model 1 with SVMs superimposed on the TDA of the gear vibration signals calculated by direct averaging. The following plots show the Model 1 with SVMs with an input of 40 gear rotation signals can correctly predict the TDA for 160 gear rotations over the entire life of the gear.

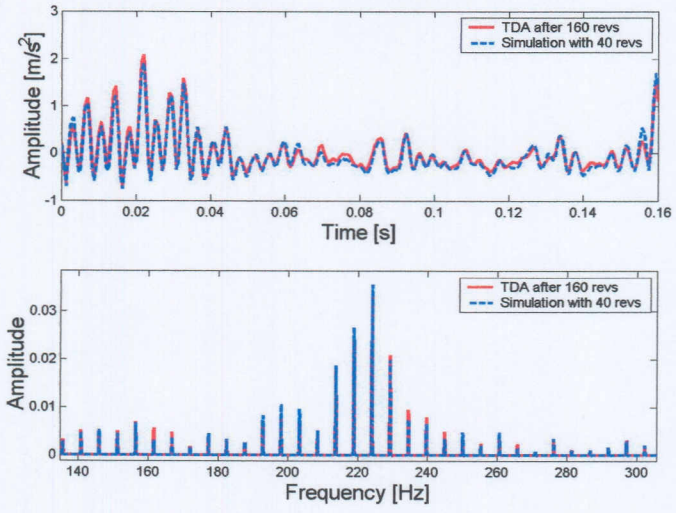


Figure 5.5 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA for 160 rotations. The measurements were taken during the running in stage of the gear.

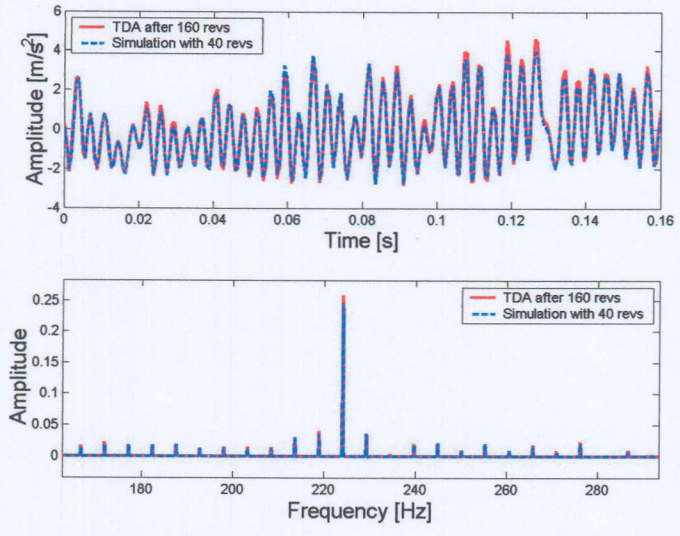


Figure 5.5 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA obtained after 160 rotations. The measurements were from the constant wear gear stage.

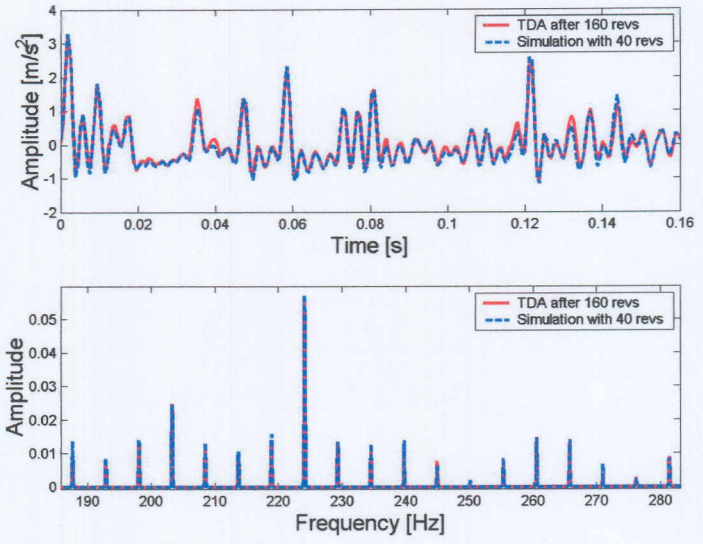


Figure 5.5 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA after 160 rotations. The measurements were taken from the wear out stage of the gear.

The FFT of the TDA calculated by direct averaging and the FFT of the simulation results are exact fits throughout the life of the gear. This indicates that Model 1 with SVMs retains the diagnostic enhancing capabilities of TDA. The good performance of SVMs is due to their superior ability to train and generalise (Gunn, 1998).

5.4 Model 2

This paragraph presents the results obtained using Model 2 from measurements taken at different stages of the gear life. The measurements were taken while the gearbox was operating under constant load conditions. Model 2 with MLP, RBF and SVMs is simulated with unseen validation data sets as presented in Table 4.4. The obtained results for each of the formulations are compared to the TDA calculated using the direct averaging for 160 gear rotations. In Model 2 all 160 rotation synchronised gear vibration signals are sequentially used in batches of 10 rotations. The rotation signals that have already been passed through the first stage of the model are deleted from the memory of the data acquisition system, while their output is saved for use in the second stage of the model. After simulation with all 160 gear rotations, there are 16 signals that will be stored in the data acquisition system. This means that the highest number of signals that will be stored in the data acquisition system is 1 batch of 10 signals and 16 outputs from the first stage of Model 2. This results in 26 signals instead of 160 rotation

synchronised gear vibration signals. This is effectively a reduction of 83.75 % of the data that needs to be stored in the data acquisition system during the time domain averaging process for this test data set.

5.4.1 Model 2 with MLP feedforward network

Figure 5.6 (a) to Figure 5.6 (c) show the prediction from Model 2 with MLP feedforward networks superimposed on TDA of the gear vibration signals calculated using direct averaging.

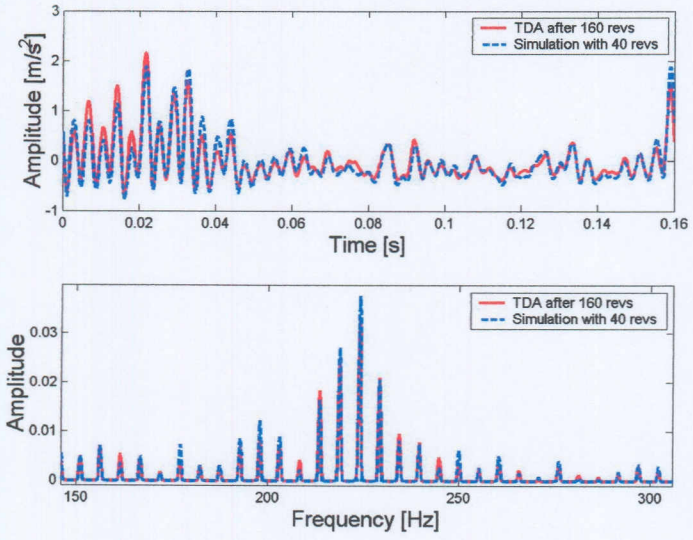


Figure 5.6 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

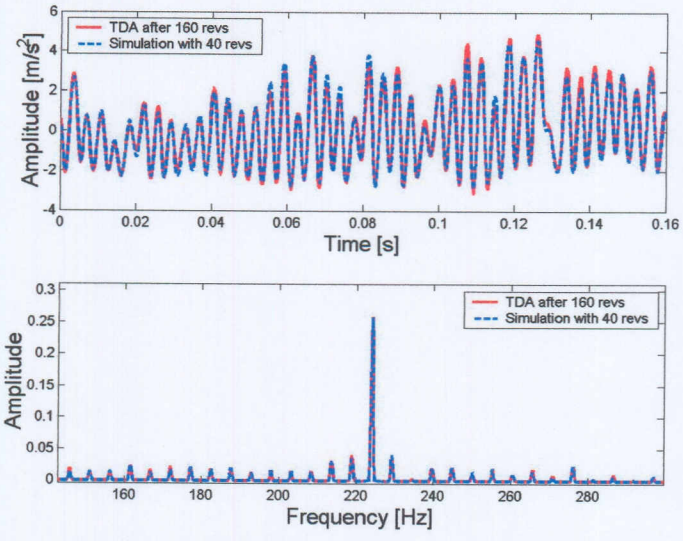


Figure 5.6 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

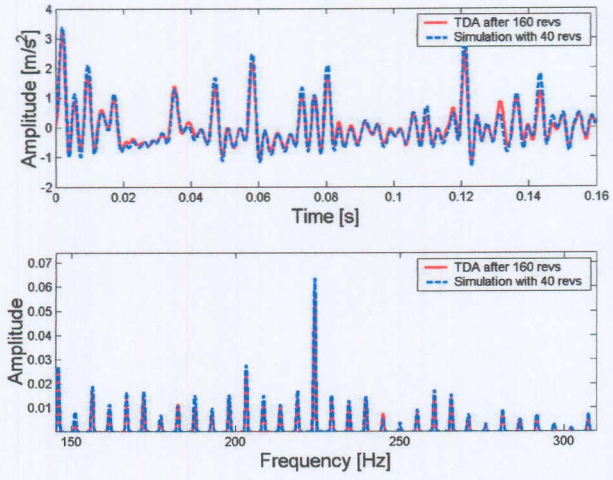


Figure 5.6 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

The above plots show in both frequency domain and time domain representation, that Model 2 with a MLP feedforward network can correctly predict the TDA for 160 gear rotations over the entire life of the gear. This is because in model 2 the network is exposed to the entire data set, therefore, it simulates more effectively.

5.4.2 Model 2 with RBF feedforward network

Figure 5.7 (a) to Figure 5.7 (c) show the prediction from Model 2 with RBF feedforward networks superimposed on the TDA of the gear vibration signals calculated using direct averaging.

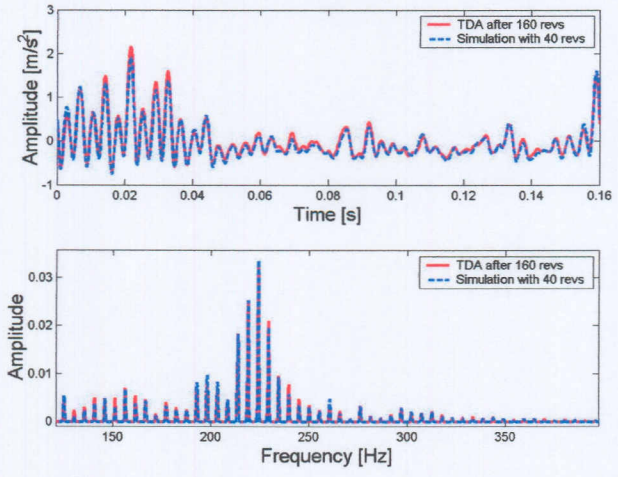


Figure 5.7 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

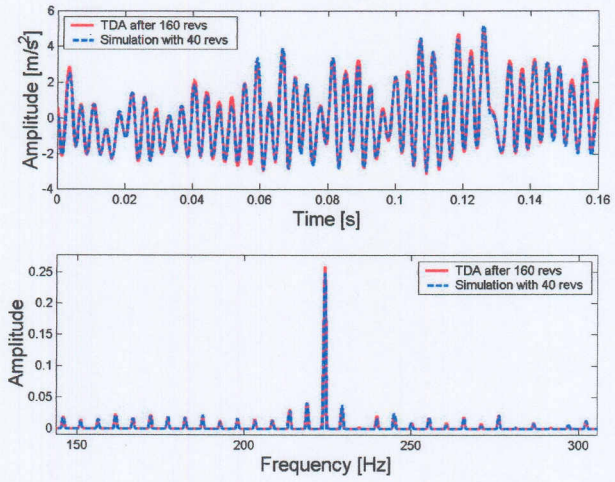


Figure 5.7 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

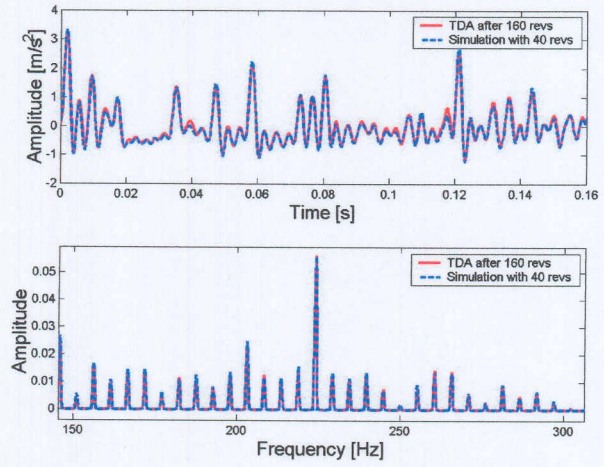


Figure 5.7 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

Figure 5.7 (a) to Figure 5.7 (c) show that Model 2 with a RBF feedforward network can correctly predict the TDA for 160 gear rotations over the entire gear life. This is because in Model 2 the network is exposed to the entire data set, therefore, it simulates more effectively.

5.4.3 Model 2 with SVMs

Figure 5.8 (a) to Figure 5.8 (c) show the prediction from Model 2 with SVMs superimposed on the TDA of the gear vibration signals calculated using the direct averaging.

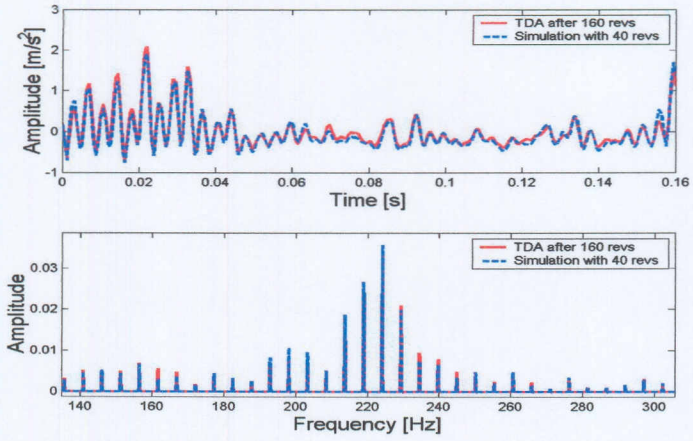


Figure 5.8 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

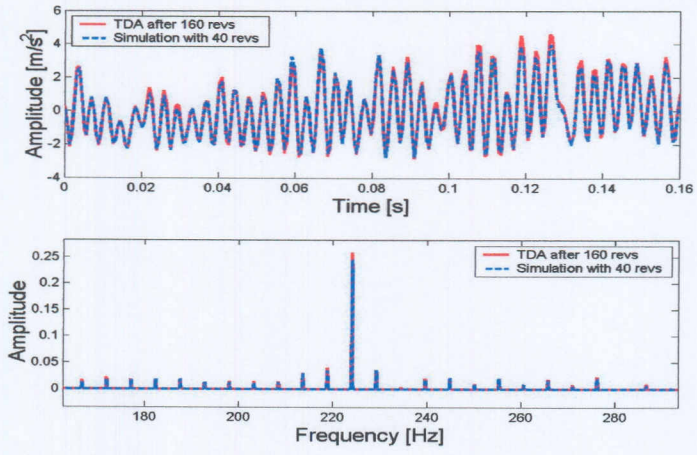


Figure 5.8 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

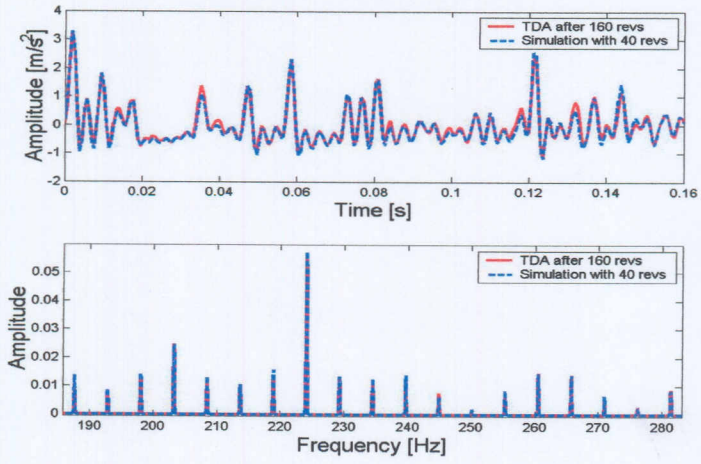


Figure 5.8 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA calculated with 160 gear rotations.

Figure 5.8 (a) to Figure 5.8 (c) show that Model 2 with a SVM feedforward network can correctly predict the TDA for 160 gear rotations over the entire life of the gear.

5.4.4 Discussion

In Figures 5.3 (a) to Figure 5.8 (a) it is observed that the TDA of the vibration from the running in stage of the gear life is fairly random. The amplitude of the TDA in the running in stage of the gear life is less than the amplitude of the TDA in the other two stages of the gear life. The random noise content of the vibration results in the presence of prominent side bands as observed frequency spectrums in Figures 5.3 (a) to Figure 5.8 (a). In Figures 5.3 (b) to Figure 5.8 (b) it is observed that the amplitude of the TDA increased and the signal is more periodic because of the constant loading condition. The reduced random noise in the signal is shown by the reduction in the side bands of the gear mesh frequency observed in the frequency spectrums in Figures 5.3 (b) to Figure 5.8 (b). Figures 5.3 (c) to Figure 5.8 (c) show an increase in the amplitude of the TDA. This is expected as one would expect the vibration would to increase as the gear fails. There is also evidence of impulse in the TDA signal. This may be because some of the gear teeth may have cracked resulting in the reduction of the meshing stiffness of those meshing tooth sets. The introduction of impulses in the TDA results in the increase in the side bands of the gear mesh frequency as observed in Figures 5.3 (c) to Figure 5.8 (c).

5.5 Assessing simulation accuracy and diagnostic capabilities

Up to this point the developed models have been used to predict the TDA of the gear vibration signal without quantifying the quality of the prediction and the diagnostic capability of the model outputs. This section presents some parameters that will be used to assess the quality of the prediction and establish whether the model predictions retain the diagnostic capabilities of the TDA.

To quantify the quality the simulation accuracy a 'fit' parameter η_{sim} (Raath, 1992) is defined. First the response error is defined as

$$e_{sim}(k) = y_{desired}(k) - y_{achieved}(k), \tag{5.3}$$

Assume that we have N data points. The simulation accuracy may then be defined as

$$\eta_{sim} = 100 \frac{\sum_{k=1}^{N} |e(k)|}{\sum_{k=1}^{N} |y_{desired}(k)|} \quad [\%] \tag{5.4}$$

The defined 'fit' parameter η_{sim} is attractive in that it gives a single value for each simulation, therefore it can be used to compare the performance of the different formulations over the entire life of the gear. A low value of 'fit' parameter η_{sim} implies a good fit while a high value implies a bad fit, therefore, a 'fit' parameter value of $\eta_{sim} = 0\%$ implies a perfect fit. In this work it was established experimentally that $\eta_{sim} = 40\%$ is a suitable upper cut-off of the simulation accuracy.

To establish whether the model predictions retain the diagnostic capabilities of the TDA two parameters are used. The first parameter is the peak value of the vibration X_{max} during a given interval T. This parameter can be used where the analyst is interested only in the overall magnitude of the vibration to distinguish between acceptable and unsatisfactory vibration states (Heyns, 2002). The second parameter is the kurtosis. The kurtosis is the fourth statistical moment of the vibration signal and it is given by

$$\text{kurtosis} = \frac{1}{\sigma^4 T} \int_0^T x^4 dt. \tag{5.5}$$

where T is a given interval, σ is the variance and x is the vibration data. The kurtosis of a signal is very useful for detecting the presence of an impulse within the signal (Norton, 1989). The peak value of the vibration X_{max} and the kurtosis of the TDA calculated using direct averaging is compared to the TDA from Model 1 and Mode 2 with all three formulations.

5.5.1 Comparison of the performance of the different formulations

The performance of the three formulations in Model 1 and Model 2 was assessed using the fit parameter to determine which of the formulations is best suited for this

application. Figure 5.9 shows the simulation accuracy η_{sim} plotted against the gear life for Model 1.

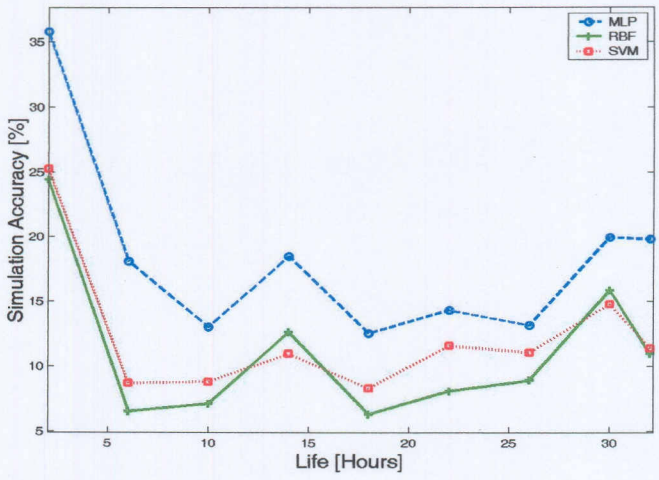


Figure 5.9 Model 1 Simulation Accuracy η_{sim} vs. gear life.

From this plot it is observed that the performance Model 1 with RBF network and Model 1 with SVMs is the same. Their performance is slightly better than the performance of Model 1 with MLP networks. The performances of all three formulations are acceptable because η_{sim} is less than the cut-off value for all the formulations.

Figure 5.10 shows the simulation accuracy η_{sim} plotted against the gear life for Model 2.

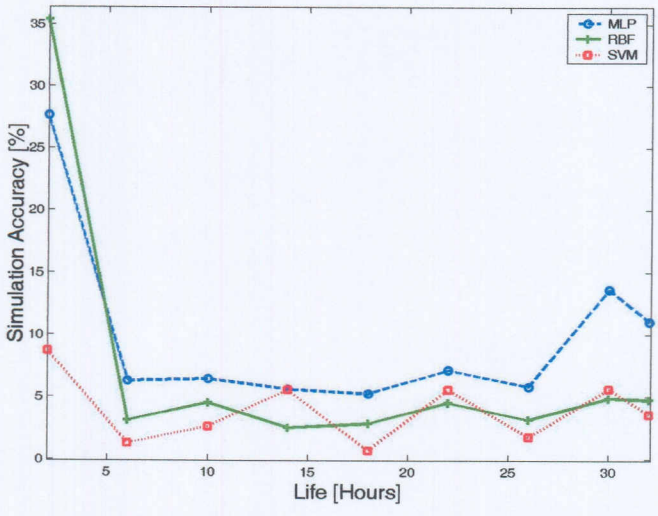


Figure 5.10 Model 2 Simulation Accuracy η_{sim} vs. gear life.

From Figure 5.10 it is observed that the performance of the formulations for Model 2 is practically the same. The performances of all three formulations are acceptable because η_{sim} is less than the cut-off value for all the formulations.

Next the performance of Model 1 and Mode 2 for each of the formulation is assessed. Figure 5.11 shows the performance of the two models with MLP feedforward networks.

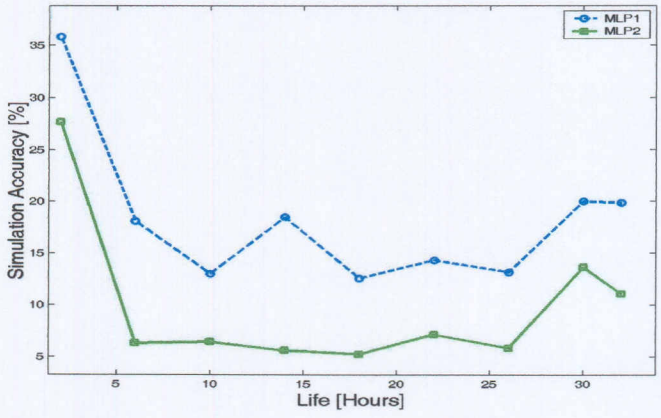


Figure 5.11 Performance of Model 1 and Model 2 with MLP feedforward networks.

From Figure 5.11 it is observed that Model 2 performs better than Model 1. This is because when simulating with Model 2 the whole data set is used as opposed to Model 1 in which only a section of the data is used.

Figure 5.12 shows the performance of the two models with RBF feedforward networks.

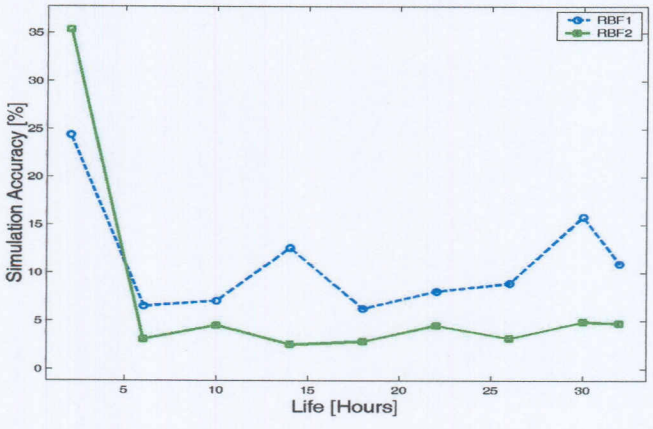


Figure 5.12 Performance of Model 1 and Model 2 with RBF feedforward networks.

Figure 5.12 shows that for RBF networks Model 1 performs better than that Model 2 in the running in stages of the gear life. After this stage the performance of Model 2 is

better than the performance of Model 1. This is because the vibration signature in the running in stages of the gear is much different to the vibration during the rest of the gear life as discussed in Section 5.4.4.

Figure 5.13 shows the performance of the two models with SVM. From this plot it is observed that Model 2 performs much better than Model 1. This is because although the vibration signatures in different stages of the gear life are different, the SVM has good generalisation properties and also the fact that when simulating with Model 2 the whole data set is used.

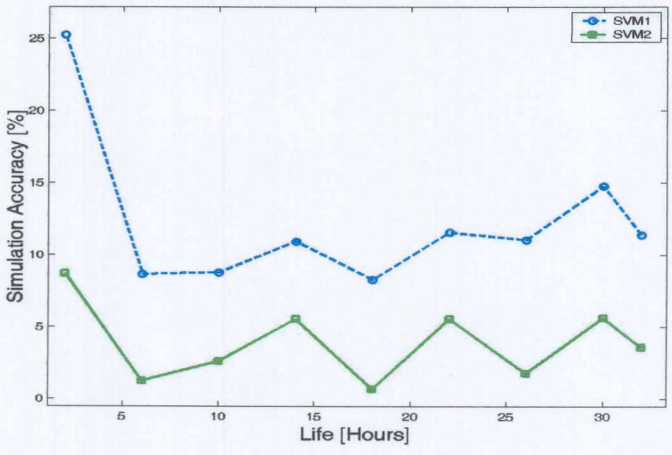


Figure 5.13 Performance of Model 1 and Model 2 with SVM.

From Figure 5.11 to Figure 5.13 it is concluded that the performances of Model 1 and Model 2 for the different formulations are quite comparable over the whole life of the gear tested under constant load conditions.

5.5.2 Comparison of the diagnostic properties of the TDA calculated by direct averaging and the TDA predicted by the developed models

To establish whether the TDA predicted by the developed models retain the diagnostic capabilities of the TDA calculated by direct averaging the peak value X_{max} and the kurtosis are used. Figure 5.14 and Figure 5.15 are plots of X_{max} and kurtosis calculated from the TDA predicted by the developed models superimposed on the X_{max} and kurtosis calculated from the TDA obtained using direct averaging for data measured under constant loading conditions. Figure 5.14 shows the results obtained using Model 1 for MLP, RBF and SVMs throughout the life of the gear.

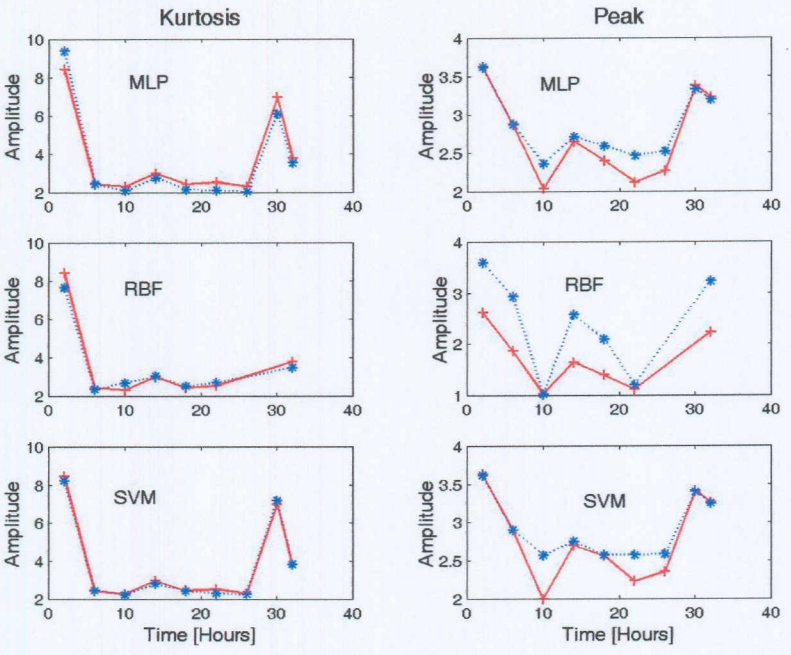


Figure 5.14 Comparison of kurtosis and peak values for the TDA calculated by direct averaging (solid line) and the TDA predicted by Model 1 (dotted line) with MLP, RBF and SVMs.

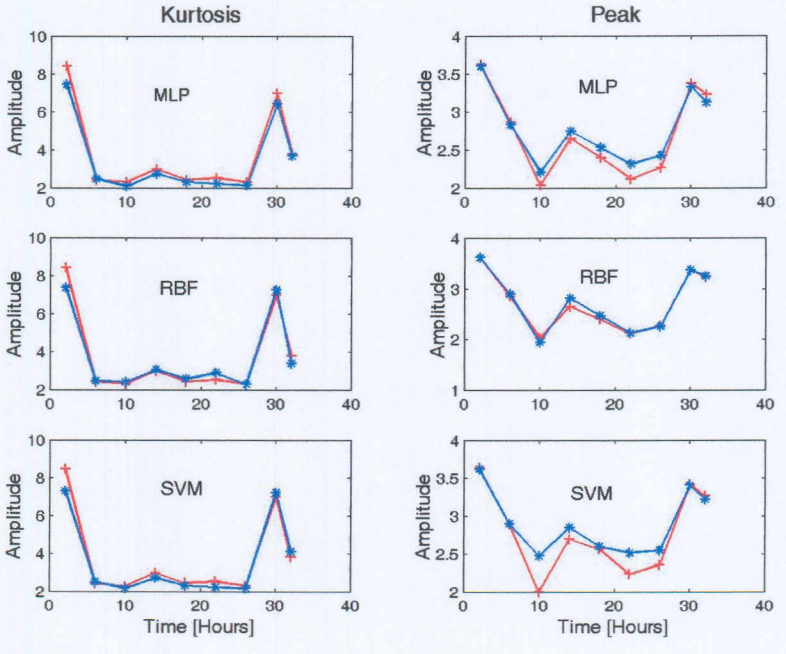


Figure 5.15 Comparison of kurtosis and peak values for the TDA calculated by direct averaging (solid line) and the TDA predicted by Model 2 (dotted line) with MLP, RBF and SVMs.

From Figure 5.14 it is observed that for all three formulations the kurtosis is an almost exact fit. This implies that the TDA predicted by Model 1 can be used to monitor the presence of impulses in the gear vibration. It is observed from this plot that there are

lots of impulses in the gear vibration during the running in and wear out stages of the gear life. On the other hand, only the peak values obtained from Model 1 with MLP and SVM are close fits and can be used to monitor the amplitude of the overall vibration. The bad performance of Model 1 with RBF is because the RBF network selected in this simulation was not optimal, therefore generalised badly to changes in the measured vibration as gear failure progressed.

Figure 5.15 shows the results obtained using Model 2 for MLP, RBF and SVMs throughout the life of the gear. It is observed from Figure 5.16 that for all three formulations the kurtosis is an exact fit therefore the TDA predicted by Model 1 can be used to monitor the presence of impulses in the gear vibration. The peak values obtained for all three formulations are close fits, therefore they can be used to monitor the amplitude of the overall vibration. The better performance on the peak values is because Model 2 uses the entire gear vibration during its simulation, therefore the network is exposed to all the underlying dynamics within the measured vibration.

5.6 Performance of developed models under varying load conditions

This topic of gearboxes operating under varying load conditions has been studied in great detail. Stander and Heyns (2001) noted the influence of varying loads on vibration monitoring of gears. Stander et al. (2002)[b] conducted an experimental investigation to observe the influence of fluctuating load conditions on the measured acceleration signal. They concluded that the load variation manifests itself as a low-frequency modulation on the measured acceleration signal. In this section the performance of the developed model for time domain averaging on data obtained from test conducted under varying load conditions is assessed. A random load with frequencies varying between 2 Hz and 5 Hz was applied. Measurements were taken at three different stages of the gear life, the running in stage, the constant wear stage and the wear out stage. The acquired data was processed as described in Chapter 4 and simulations were done using unseen validation sets for Model 1 and Model 2.

5.6.1 Simulations with Model 1

This paragraph presents the results obtained using Model 1 for test data obtained from different stages of the gear life of tests conducted under varying load conditions. Figure

5.16 (a) to Figure 5.16 (c) show the results obtained when Model 1 with MLP network is simulated with a validation set of 40 unseen gear rotations for the three different gear life stages.

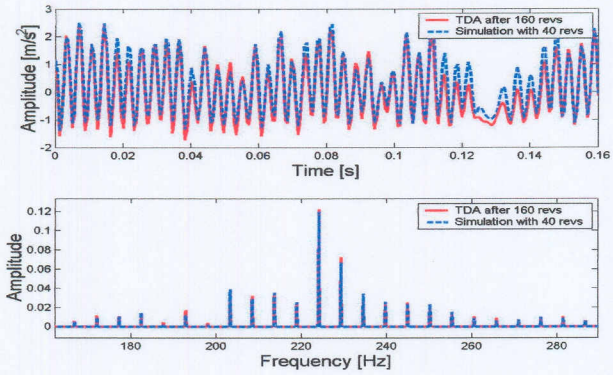


Figure 5.16 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken during the running in stage of the gear.

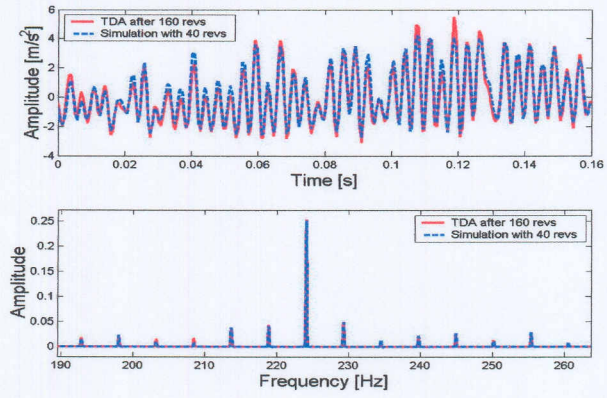


Figure 5.16 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken from the constant wear stage of the gear.

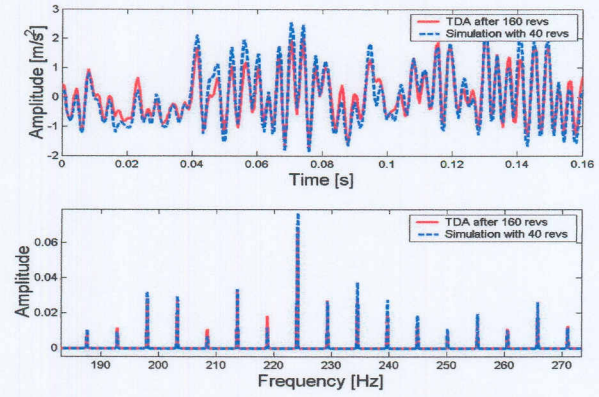


Figure 5.16 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken in the wear out stage of the gear.

The above plots show that Model 1 with 40 gear rotations can predict the TDA for 160 gear rotations fairly well over the different life stages of the gear under varying load conditions. This is because of the good generalisation capabilities of the MLP network. The frequency spectrum in this plot indicates that Model 1 with MLP can pick up the side bands of the gear mesh frequency.

Figure 17 (a) to Figure 17 (c) show the results obtained when Model 1 with RBF network is simulated using 40 unseen gear rotations for the three different gear life stages.

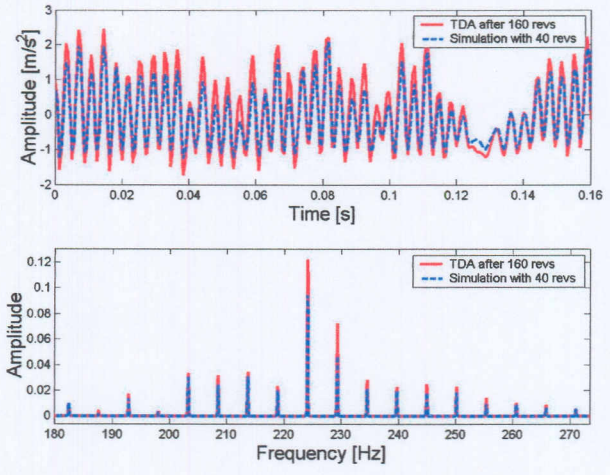


Figure 5.17 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken during the running in stage of the gear.

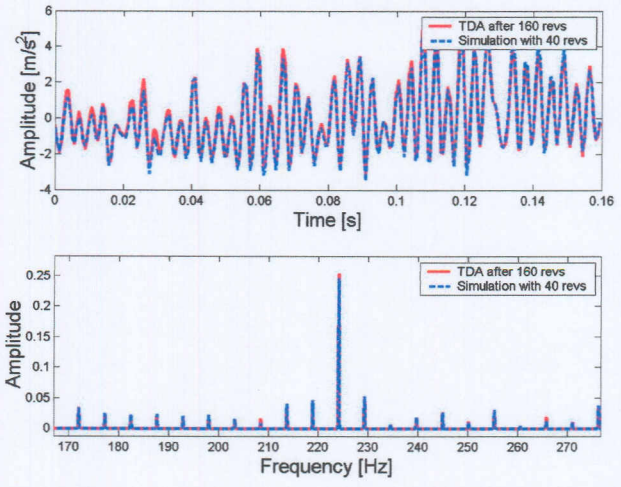


Figure 5.17 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken from the constant wear stage of the gear.

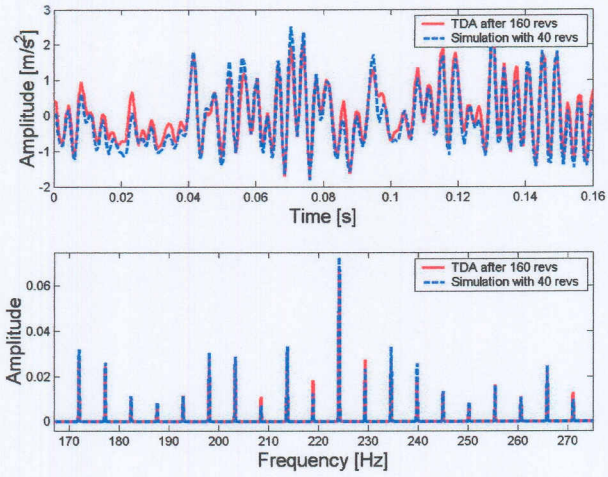


Figure 5.17 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken in the wear out stage of the gear.

From above plots it is observed that Model 1 with RBF and 40 gear rotations can predict the TDA for 160 gear rotations fairly well but the prediction of the running in stage of the gear life is poor. This is because the vibration signature of the gear vibration are different and the generalisation capabilities of the selected RBF architecture was not as good as that of MLP network.

Figure 18 (a) to Figure 18 (c) show the results obtained when Model 1 with SVM is simulated using 40 unseen gear rotations for the three different gear life stages.

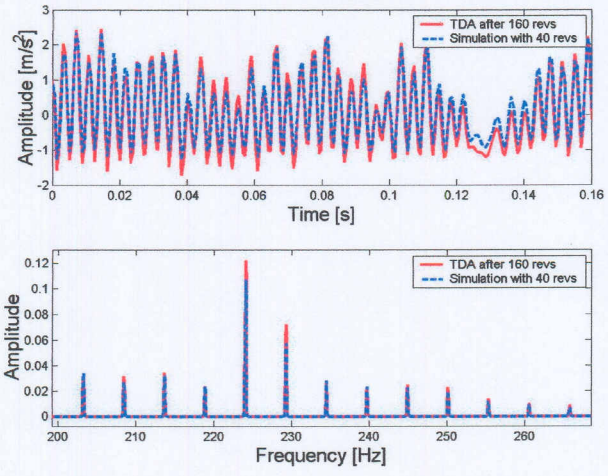


Figure 5.18 (a) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken during the running in stage of the gear.

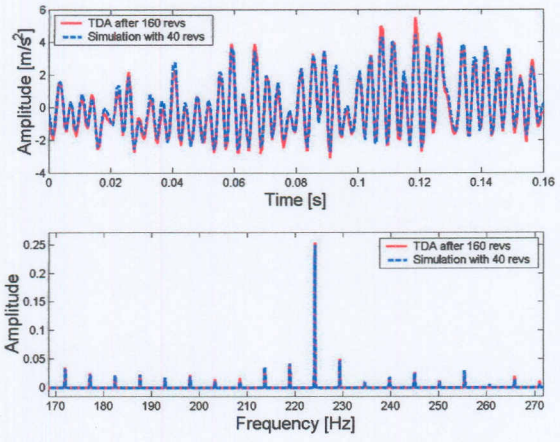


Figure 5.18 (b) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken from the constant wear stage of the gear.

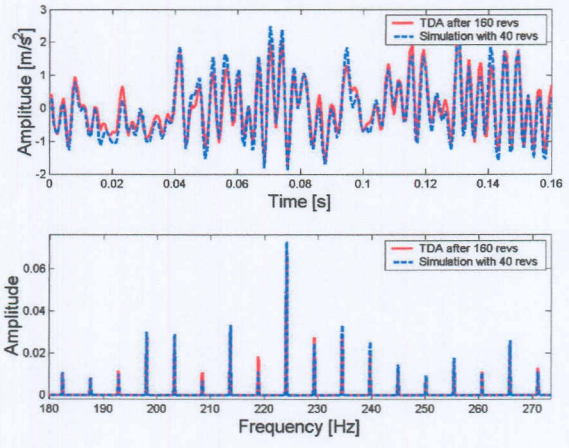


Figure 5.18 (c) Model 1 prediction with validation set of 40 gear rotations superimposed on the TDA from 160 rotations from measurements taken in the wear out stage of the gear.

Model 1 with SVMs produces poor results for the running in stage of the gear life. This is because the vibration signatures of the gear vibration are different in the different stages and the SVM does not generalise well enough.

5.6.2 Simulations with Model 2

In this paragraph Model 2 simulations with unseen validation data from different stages of the gear life are presented.

Figure 19 (a) to Figure 19 (c) show the results obtained when Model 2 with a MLP network is simulated with unseen gear rotations for the three gear life stages.

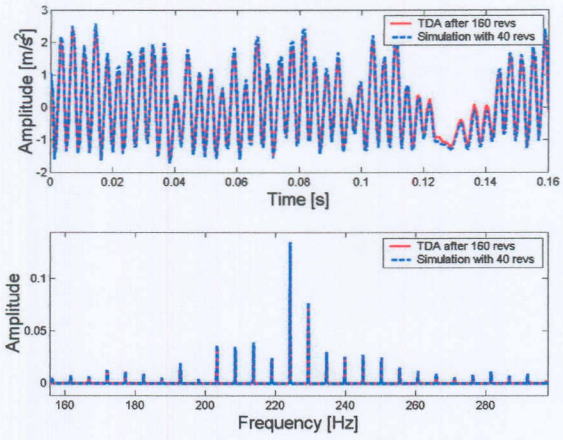


Figure 5.19 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA obtained after 160 gear rotations

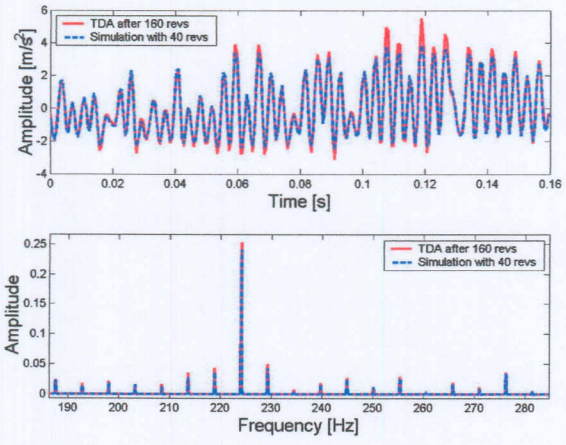


Figure 5.19 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA obtained after 160 gear rotations

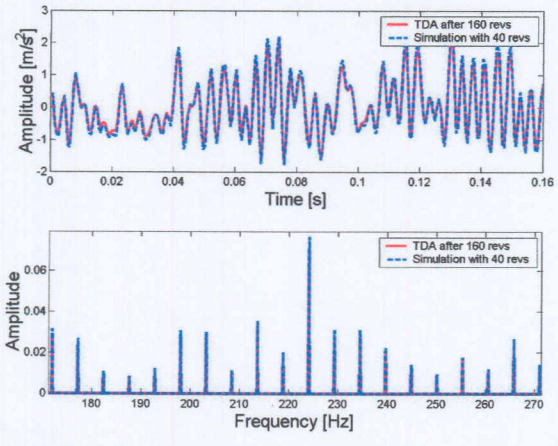


Figure 5.19 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA obtained after 160 gear rotations

Figure 5.19 (a) to Figure 5.19 (c) show that Model 2 with a MLP feedforward network can correctly predict the TDA for 160 gear rotations over the entire life of the gear. This is because Model 2 uses the whole data set as opposed to Model 1 that uses only a section of the data set.

Figure 20 (a) to Figure 20 (c) show the results obtained when Model 2 with a RBF network is simulated with unseen gear rotations for the three gear life stages.

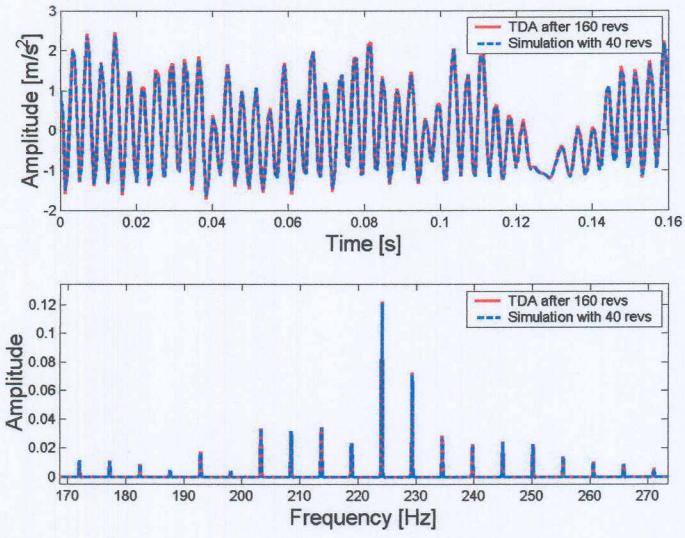


Figure 5.20 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA obtained after 160 gear rotations

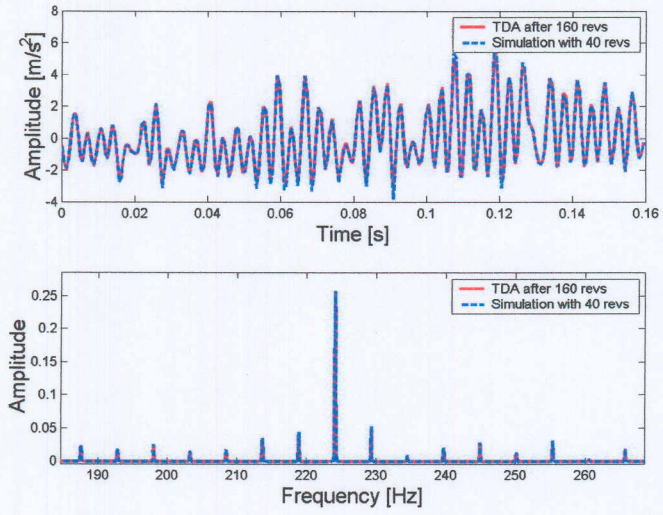


Figure 5.20 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA obtained after 160 gear rotations

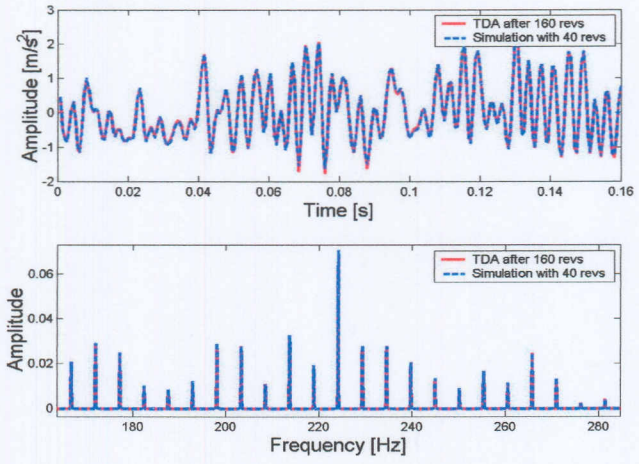


Figure 5.20 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA obtained after 160 gear rotations

Figure 5.20 (a) to Figure 5.20 (c) show that Model 2 with a RBF feedforward network can correctly predict the TDA for 160 gear rotations over the entire life of the gear. The good performance can again be attributed to the fact that Model 2 uses the whole data during simulation as opposed to Model 1 that uses only a section of the data set.

Figure 21 (a) to Figure 21 (c) show the results obtained when Model 2 with SVMs is simulated with unseen gear rotations for the three gear life stages.

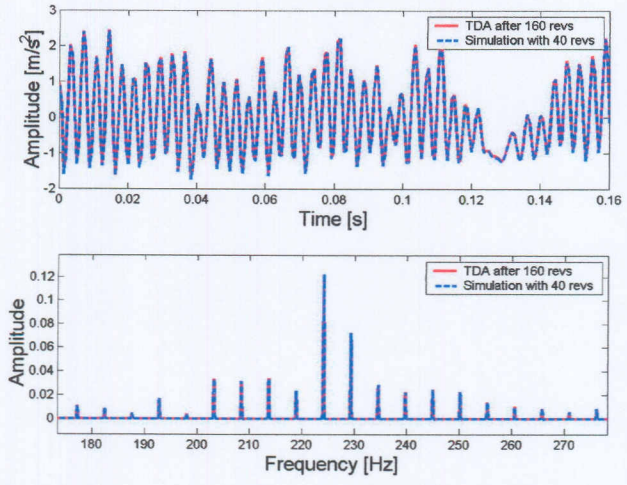


Figure 5.21 (a) Model 2 prediction with a validation set measured during the running in stage of the gear life superimposed on the TDA obtained after 160 gear rotations

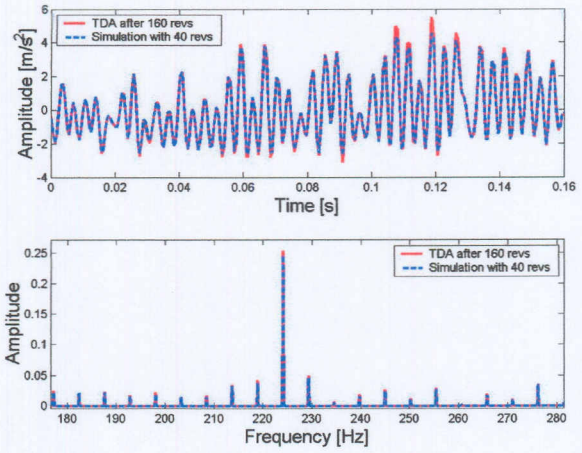


Figure 5.21 (b) Model 2 prediction with a validation set measured during the constant wear stage of the gear life superimposed on the TDA obtained after 160 gear rotations

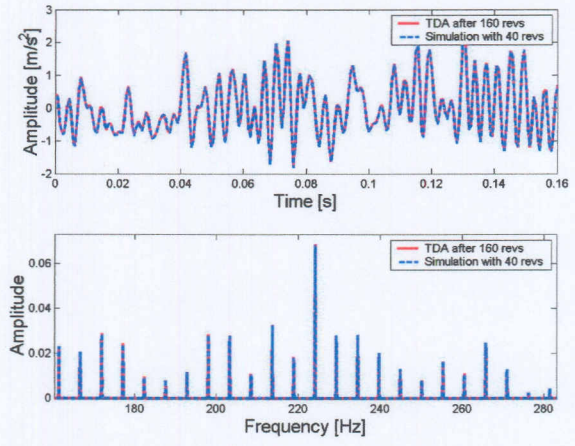


Figure 5.21 (c) Model 2 prediction with a validation set measured during the wear out stage of the gear life superimposed on the TDA obtained after 160 gear rotations

Figure 5.21 (a) to Figure 5.21 (c) show that Model 2 with a SVM network correctly predicts the TDA for 160 gear rotations over the entire life of the gear. The good performance is attributed to the fact that Model 2 uses the whole data during simulation as opposed to Model 1 that uses only a section of the data set.

It is observed from the above plots that ANNs and SVMs can correctly predict the TDA under varying load conditions. This is because when properly trained, ANNs and SVMs can map nonlinearity between an input and output space with good generalisation. Secondly, the applied load was random, therefore, the load modulation on the vibration signature was not synchronous with the vibration of the shaft. This load modulation

condition is called non-synchronous load modulation (Stander and Heyns, 2003) and the TDA can suppress load modulation under non-synchronous fluctuating load conditions because of its randomness relative to the rotation of the gear.

5.6.3 Comparison of the performance of the different formulations under varying load

The performance of the three formulations in Model 1 and Model 2 was assessed using the fit parameter to determine which of the formulations is best suited for this application. Figure 5.22 and Figure 5.23 shows the simulation accuracy η_{sim} plotted against the gear life for Model 1 and Model 2, respectively.

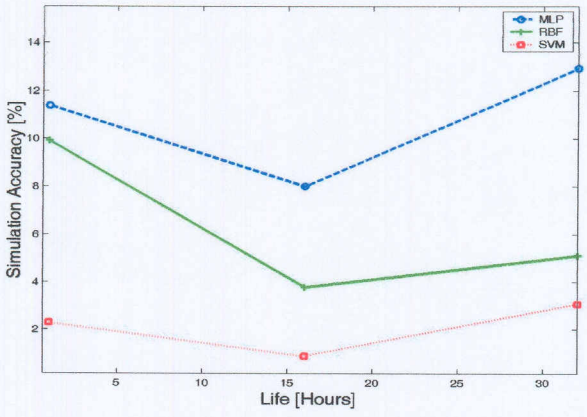


Figure 5. 22 Model 1 Simulation Accuracy η_{sim} vs. gear life

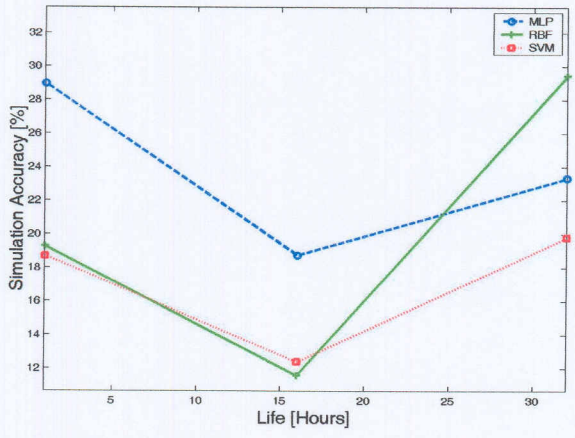


Figure 5. 23 Model 2 Simulation Accuracy η_{sim} vs. gear life

For Model 1 it is observed that SVMs perform best for varying load conditions and MLP gives the worst performance. This is because of the structural risk minimisation used in SVMs, which is said to generalise better than the empirical risk minimisation

used in neural networks (Vapnik 1995; Gunn, 1998). For Model 2 the performance of the formulation is the same. This is because Model 2 uses the whole vibration during simulation therefore the network is exposed to all the transient effects within the data.

The following set of plots compare the performance Model 1 and Model 2 for the different formulations.

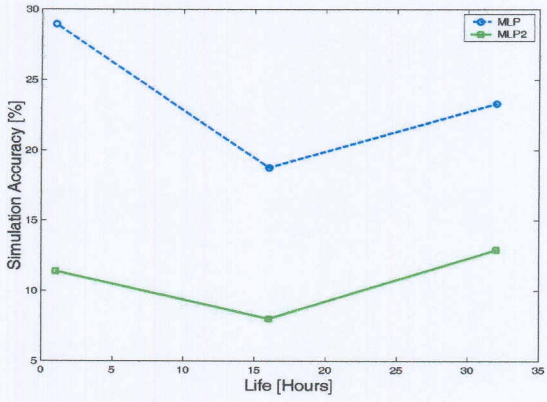


Figure 5.24 Performance of Model 1 and Model 2 with MLP feedforward networks

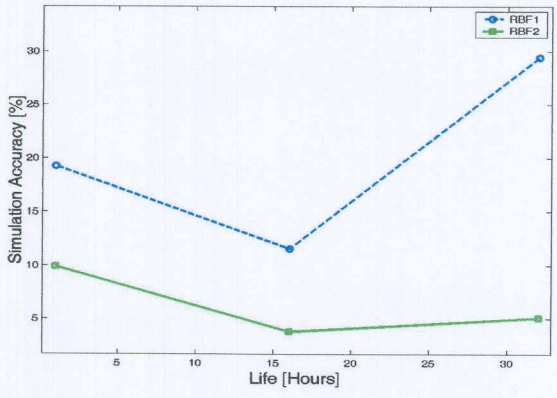


Figure 5.25 Performance of Model 1 and Model 2 with RBF feedforward networks

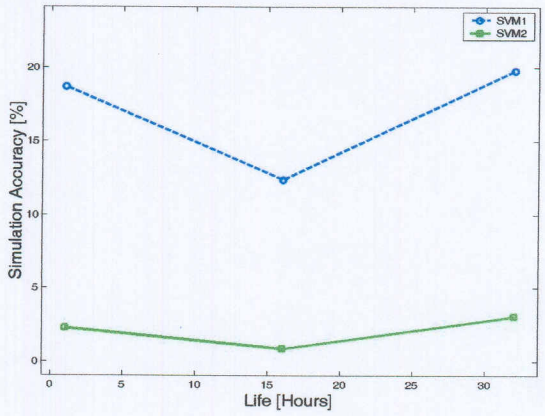


Figure 5.26 Performance of Model 1 and Model 2 with SVM

The above plots show that for varying load condition Model 2 performs much better than Model 1 for all three formulations. This is due to the fact that although Model 2 only uses 10 inputs at a time, it still uses the whole data for simulation as opposed to Model 1 that uses only a section of the data set for simulation. This allows Model 2 to train and simulate more efficiently since it is exposed to all the underlying dynamics within the data set.

5.6.4 Comparison of the diagnostic properties of the TDA calculated by direct averaging and the TDA predicted by the developed models

This paragraph presents the X_{max} and kurtosis to establish whether the TDA predicted by the developed models retains the diagnostic capabilities of the TDA calculated by direct averaging the peak value X_{max} and the kurtosis are used. Figure 5.27 and Figure 5.28 are plots of X_{max} and kurtosis calculated from the TDA predicted by the developed models superimposed on the X_{max} and kurtosis calculated from the TDA calculated using direct averaging for data measured under varying load conditions.

Figure 5.27 shows the results obtained using Model 1 for MLP, RBF and SVMs. In Figure 5.27 and Figure 5.28 X_{max} and kurtosis are only plotted for the running in, constant wear and wear out stages of the gear life. It is observed that the kurtosis is not a good fit. This implies that Model 1 cannot be used to monitor the presence of impulses in the gear vibration. The peak values for TDA calculated using direct averaging and the peak values for TDA obtained using Model 1 fit well. This implies that the peak values of the TDA obtained using Model 1 can be used to monitor the overall vibration of the gear signal.

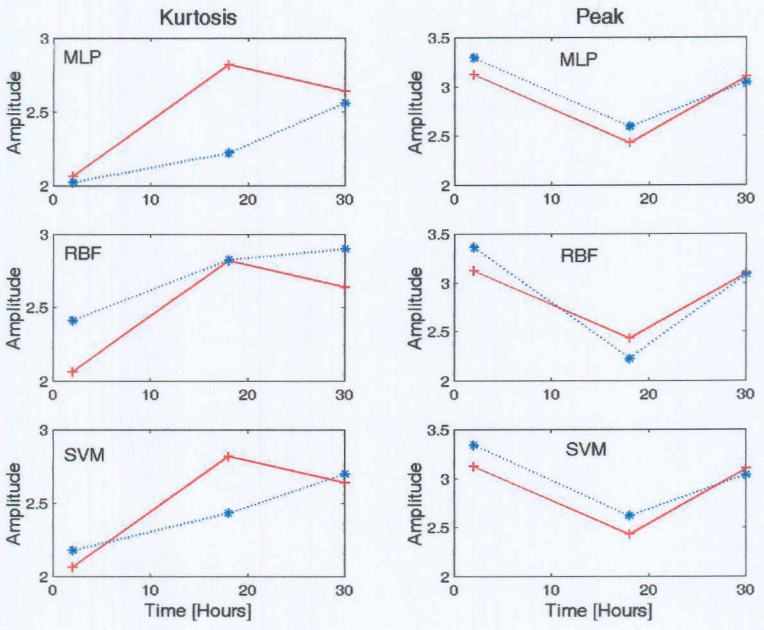


Figure 5.27 Comparison of kurtosis and peak values for the TDA calculated by direct averaging (solid line) and the TDA predicted by Model 1 (dotted line) with MLP, RBF and SVMs.

Figure 5.28 shows the results obtained using Model 2 for MLP, RBF and SVMs during the running in, constant wear and wear out stages of gear life.

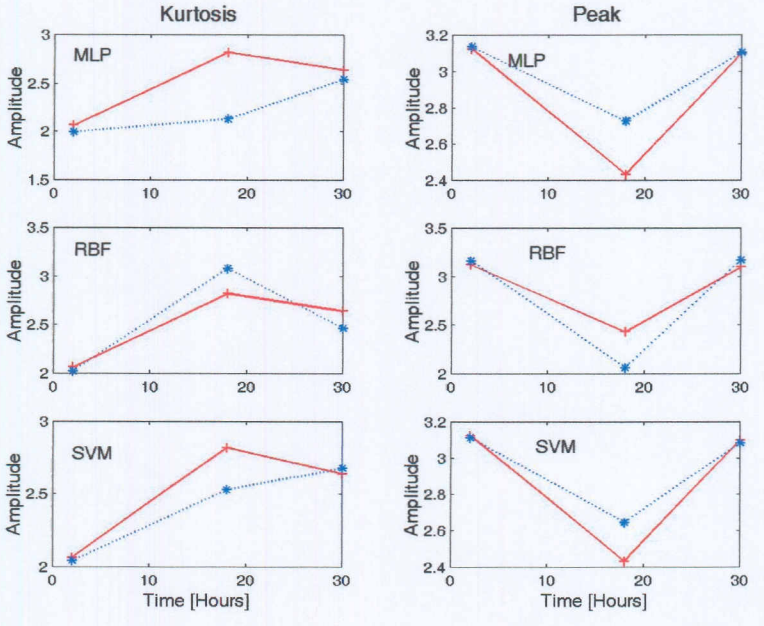


Figure 5.28 Comparison of kurtosis and peak values for the TDA calculated by direct averaging (solid line) and the TDA predicted by Model 1 (dotted line) with MLP, RBF and SVMs.

It is observed from Figure 5.28 that both the kurtosis and the peak values are not far off. This implies that the TDA predicted by Model 2 can be used to monitor impulses and the overall gear vibration. The superior performance of Model 2 is attributed to the fact that Model 2 exposes the formulations to the entire vibration during the simulation process. Table 5.2 and Table 5.3 present a summary of the properties of Model 1 and Model 2 for the three formulations.

Table 5.2 Summary of properties for Model 1 with MLP, RBF and SVMs.

	Strength	Weaknesses	Ideal application
MLP	Good generalisation under constant load conditions Good generalisation under varying load conditions	Depends on training and generalisation of selected network	Monitoring of overall vibration and impulses in a gear vibration under constant load conditions Monitoring of peak values under varying load conditions
RBF	Good generalisation under constant load conditions	Poor generalisation under varying load conditions Depends on generalisation of selected network	Monitoring of overall vibration and impulses constant loading conditions Monitoring of peak values under varying load conditions
SVM	Good generalisation under constant load conditions Good generalisation under varying load conditions	Depends on generalisation of SVM	Monitoring of overall vibration and impulses under constant loading conditions Monitoring of peak values under varying load conditions

Table 5.3 Summary of properties for Model 2 with MLP, RBF and SVMs

	Strength	Weaknesses	Ideal application
MLP	Good generalisation under constant load conditions Good generalisation under varying load conditions	Looses diagnostic capability for overall vibration under varying loads Depends on generalisation of selected network	Monitoring of overall vibration and impulses in under both constant and varying load conditions
RBF	Good generalisation under constant load conditions Good generalisation under varying load conditions	Poor generalisation under varying load conditions Looses diagnostic capability for overall vibration under varying loads Depends on generalisation of selected network	Monitoring of overall vibration and impulses in under both constant and varying load conditions
SVM	Good generalisation under constant load conditions Good generalisation under varying load conditions	Depends on generalisation of SVM	Monitoring of overall vibration and impulses in under both constant and varying load conditions

5.7 Conclusion

In this chapter the synchronous filter for time domain averaging of gear vibration data developed in Chapter 4 is tested on a new vibration data set from the accelerated gear life test rig to assess its suitability for use over the entire life of the gear. For measurements from tests carried out under constant load conditions the performances of Model 1 and Model 2 are practically the same over the entire life of the gear. For measurements from a test carried out under varying load conditions Model 2 performs better than Model 1 over the entire life of the gear. The superior performance of Model 2 is because Model 2 uses the whole data set for training and simulation as opposed to Model 1, which uses only a section of the data set. Using the whole data set during training and simulation exposes the formulations in the model to all transient effects within the data resulting in a more accurate TDA prediction. The performance of Model 1 strongly depends on the generalisation capabilities of the formulation that is used.

Chapter 6

Conclusions and recommendations for future work

6.1 Conclusions

The monitoring of incipient failure in gears and gearbox systems is of outmost importance to engineers since failures in large gearboxes without any backup systems could easily lead to production losses that can quickly amount to millions of Rands. Most of the vibration analysis techniques for gearboxes require time domain or synchronous averaging. Although direct time domain averaging has been around for decades, it still remains a challenge to develop an efficient time domain averaging filter (synchronous filter) that is suitable for implementation on an online gear condition monitoring system. This is particularly so because direct time domain averaging requires a large amount of data to be collected and stored in the data acquisition system before the TDA is calculated.

In this work the use of ANNs and SVMs in the development of a synchronous filter for time domain averaging of gear vibration data is investigated. Two models that utilise ANNs and SVMs for time domain averaging are developed. The first model (Model 1) utilises a feedforward network structure to map the input space (rotation synchronised gear vibration signals) to the target (TDA of the rotation synchronised gear vibration signals) in a single step. The second model (Model 2) estimates the TDA of the input space in small sequential steps, analogous to taking a running average of the input space. Model 2 consists of a number of small feedforward networks but instead of the networks being used to predict the TDA of the whole input space in one step, are used to first sequentially predict the average of subsections of the input space (instantaneous averages). The output of the first set of feedforward networks are used as inputs into a second feedforward network that predicts the TDA of the whole input space.

The developed models considerably reduce the amount of gear vibration data that needs to be stored in the data acquisition system to execute synchronous averaging. This characteristic of the developed synchronous filtering models brings us yet another step

closer to the development of an online vibration based gearbox condition monitoring system that makes use of TDA to enhance its diagnostic capabilities.

Chapter 1 of this work presents a literature study of different vibration based gear condition monitoring techniques, filtering and the application ANNs and SVMs in pattern recognition. ANNs and SVMs are attractive in study because of the following properties:

- They can form arbitrary decisions so that any complex mapping from a set of noise-contaminated signals to a noise-free signal can be realized.
- They can easily be implemented as software or in specialized hardware
- They are quite resilient against distortions in the input data and have a capability to learn and generalize well when they are properly trained.

In Chapter 2 the theory and mathematics of existing time domain averaging models are presented. The characteristics of some of the existing time domain averaging models are investigated using numerical examples and experimental vibration data from the accelerated gear life test rig. From this analysis it is concluded that calculating the TDA by direct averaging is most suitable for filtering out the vibration content that is not synchronous with the rotation of the gear of interest.

In Chapter 3 the theories of the MLP neural network, the RBF neural network and SVMs are presented. Preliminary simulations using data from the accelerated gear life test rig to investigate the suitability of these formulations for use in the synchronous filter are conducted. It is concluded from the simulation results that MLP, RBF and SVMs are all suitable for use in the synchronous filtering model.

Chapter 4 presents the actual development process for the synchronous filter for time domain averaging of gear vibration data. Two different filter models are developed. Using Model 1, which utilises ANN or SVMs to map input space (rotations synchronised gear vibration signals) to the target (TDA calculated with 160 rotation synchronised gear vibration signals by direct averaging) in one step an input vibration data reduction of 75 % was achieved. At first glance SVMs seem to be a more attractive option because of the slightly better TDA prediction they produce. Their attractiveness

is reduced by the fact that they are more computationally expensive than MLP and RBF, therefore the analyst needs to be cautious when SVMs are implemented in an online system that requires retraining regularly. On the other hand the MLP and RBF networks are much quicker and easier to train, therefore can be suitable for online systems even when required to retrain online. Model 2 operates in two stages. In the first stage it uses 10 inputs (10 rotations synchronised gear vibration signals) to predict the instantaneous TDA of the gear vibration. The input data to the first stage can immediately be deleted from the memory of the data acquisition system after use. The outputs of the first stage are saved and used as inputs to a second feedforward network to predict the TDA of the entire gear vibration signal. In this work this results in an effective data reduction of 83.75 % in the amount of data that needs to be stored in the data acquisition system in order to calculate the TDA.

In Chapter 5 the developed synchronous filters for time domain averaging of gear vibration data are tested on a new data set from the accelerated gear life test rig to assess their diagnostic capabilities and suitability for use over the entire gear life. The performances of the different formulations on the developed models are found to be equal for data measured under constant load conditions for both Model 1 and Model 2. For data measured under varying load conditions Model 2 performs much better than Model 1 over the entire gear life for all three formulations. This is because Model 2 uses the whole data set for training and simulation therefore it exposes the used formulation to all the transient effects within the data set. The performance of Model 1 depends on the generalisation capabilities of the formulation that is used.

6.2 Recommendations for future work

As future work on the development of a synchronous filter for time domain averaging of gear vibration data the following issues still need to be addressed:

- The two models that were proposed in this work should be made more robust by implementing proper optimisation schemes for the network architecture for a specific data set.
- The developed models should be implemented in a DSP board for use in actual online gear condition monitoring system.

- The test data that was used to test the developed models was from a controlled experimental set-up. The developed models should be tested on data from an industrial application.
- An investigation of the suitability of the developed models for use in condition monitoring of other rotating machinery that require time domain averaging should be conducted.

References

Alsing, S. G., Bauer Jr., K. W. and Miller, J.O., (2002) A multinomial selection procedure for evaluating pattern recognition algorithms, Pattern Recognition 35, pp 2397 – 2412.

Aronszajn, N., (1950) Theory of reproducing kernels, Trans. Amer. Math. Soc. 686, pp 337- 404.

Augustyn, G. L., Batko, W. and Wierzbicki, J., (2003) Contex Filtering, Tenth International Congress on Sound and Vibration 7-10 July. Stockholm, Sweden, pp 4213-4219.

Baydar, N. and Ball, A., (2000) Detection of gear deterioration under varying load conditions by using the Instantaneous Power spectrum, Mechanical Systems and Signal Processing 14(6), pp 907-921.

Baydar, N. and Ball, A., (2001) A comparative study of acoustic and vibration signals in detection of gear failure using Wigner-Ville Distribution, Mechanical Systems and Signal Processing 15(6), pp 1091-1107.

Bishop, C. M. and Nabney., I.T., (1996) Netlab neural network software, http:/www.ncrg.aston.ac.uk/netlab.

Bishop, C.M., (1995) Neural networks for pattern recognition, Oxford University Press, Oxford, UK.

Braun S. and Seth, B., (1980) Analysis of repetitive mechanism signatures, Journal of Sound and Vibration 70(4), pp 513-526.

Braun, S., (1975) The extraction of periodic waveforms by time domain averaging, Acustica 32, pp 69-77.

Bringham, E.O., (1974) The Fast Fourier Transform. Englewood Cliffs, NJ: Prentice-Hall.

Broomhead, D.S. and Lowe, D., (1988) Multivariable functional interpolation and adaptive networks, Complex Systems 2, pp 321-355.

Burges, C. J. C., (1998) A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery (2), pp 121-167.

Davel, J. G., (2003) Correlation between vibration levels and expected life of cylindrical gears, BEng (Mech). Final year project, University of Pretoria.

Decker, H.J., (2002)[a] Crack detection for aerospace quality spur gears, *NASA/TM*, 2002-211492.

Decker, H.J., (2002)[b] Gear crack detection using tooth analysis, *NASA/TM*, 2002-211491,

Dempsey, P.J. and Afjeh, A.A., (2002) Integrating oil debris and vibration gear damage detection technologies using fuzzy logic, *NASA/TM*, 2002-211126.

Dempsey, P.J., Handschuh, R.F. and Afjeh, A.A., (2002) Spiral bevel gear damage detection using decision fusion analysis, *NASA/TM*, 2002-211814.

Fidêncio, P. H., Poppi, R.J. and de Andrade, J. C., (2002) Determination of organic matter in soils using radial basis function networks and near infrared spectroscopy, Analytica Chimica Acta 453, pp125-134.

Gardner, M.W. and Dorling, S.R., (1999) Neural network modelling and prediction of hourly NO_x and NO_2 concentrations in urban air in London Atmospheric Environment 33, pp 709-719.

Gaudart, J., Giusiano, B. and Huiart, L., (2002) Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data, Computational Statistics & Data Analysis.

Geman, S., Bienenstock, E., and Doursat, R., (1992) Neural networks and the bias/variance dilemma, Neural Computation 4, pp 1-58.

Gunn, S. R., (1998) Support vector machines for classification and regression, Technical Report, Department of Electronics and Computer Science, University of Southampton.

Gunn, S. R., Brown, M. and Bossley, K. M., (1997). Network performance assessment for neurofuzzy data modelling. In Liu, X., Cohen, P., and Berthold, M., editors, Intelligent Data Analysis, volume 1208 of Lecture Notes in Computer Science, pp 313-323.

Gunn, S., (1998) Matlab SVM Toolbox, http:/www.isis.ecs.soton.ac.uk/ resources/ svminfo

Haykin, S., (1999) Neural networks, 2nd edition, Prentice-Hall Inc, New Jersey, USA.

Heyns, P.S., (2002) Mechanical vibrations measurement and analysis, MEV 732 course notes, University of Pretoria.

Hinton, G.E., (1987) Learning translation invariant recognition in massively parallel networks, In J.W. de Bakker, A.J. Nijman, and P.C. Treleaven (Eds.), Proceedings PARLE Conference on Parallel Architectures and Languages Europe, pp. 1-13. Berlin: Springer-Verlag.

Hopfield, J.J., (1987) Learning algorithms and probability distributions in feed-forward and feed-back networks, Proceedings of the National Academy of Science, Vol. 84, 8429-8433.

MacKay D. J. C., (1994) Bayesian non-linear modelling for the energy prediction competition, ASHRAE Transactions 100(2), pp 1053-1062.

Marwala, T., (2001) Fault identification using neural networks and vibration data, Ph.D. Thesis, University of Cambridge.

McFadden, P.D. and Smith, J.D., (1985) A signal processing technique for detecting local defects in a gear from the signal average of the vibration, Proceeding of the Institute of Mechanical Engineers 199(C4), pp 287-292.

McFadden, P.D., (1986) Detecting fatigue cracks in gears by amplitude and phase nodulation of the meshing vibration, American Society of Mechanical Engineers, Journal of Vibration Acoustics Stress and Reliability in design 199(2), pp 165-170.

McFadden, P.D., (1987) A revised model for the extraction of periodic waveforms by time domain averaging, Mechanical Systems and Signal Processing 1, pp 83-95.

McFadden, P.D., (1987) Examination of a technique for the early detection of failure in gears by signal processing of the time domain average of the meshing vibration, Mechanical Systems and Signal Processing 1, pp 173-183.

McFadden, P.D., (1989), Interpolation techniques time domain averaging of gear vibration, Mechanical Systems and Signal processing 3(1), pp 87-97.

McFadden, P.D., Cook, J.G. and Forster, L.M, (1999) Decomposition of gear vibration signals by the Generalised S Transforms, Mechanical Systems and Signal Processing 13(5), pp 691-707.

Mdlazi, L., Marwala, T., Stander, C., Scheffer, C. and Heyns P.S. (2003) Principal component analysis and Automatic relevance determination for damage identification in structure, Proceedings of the 21st International Modal Analysis Conference, San Antonio, pp 37-42.

Moczulski, W., (1987) The digital synchronous filtering technique, Mechanical Systems and Signal Processing 1, pp 197-210.

Møller, M., (1993) A scaled conjugate gradient algorithm for fast supervised learning, Neural Networks, Vol 6, pp 525-533.

Moody, J. and Darken, C.J., (1989) Fast learning of networks of locally-tuned processing units, Neural Computation 1 (2), pp 281-294.

Neal, R. M., (1996) Bayesian Learning for Neural Networks. New York, NY: Springer Verlag.

Neal, R. M., (1998) Assessing relevance determination methods using DELVE, Neural Networks and Machine Learning. New York, NY: Springer-Verlag.

Norton, M. P., (1989) Fundamentals of noise and vibration analysis for engineers, New York: Cambridge University Press.

Paya, B.A., Esat, I.I. and Badi, M.N.M., (1997) Artificial Neural Networks based fault diagnostics of rotating machinery using Wavelet Transforms as a pre-processor, Mechanical Systems and Signal Processing 11(5), pp 751-765.

Powell, M. J. D., (1987) Radial basis functions for multivariable interpolation: a review. In. Mason, J. C. and Cox M. G., Algorithms for approximation, pp 143-167. Oxford: Clarendon Press.

Raath, A.D., (1992) Structural dynamic response reconstruction in the time domain, PhD thesis, Department of Mechanical and Aeronautical Engineering, University of Pretoria.

Ramesh, R., Mannan, M.A., Poo, A.N. and Keerthi, S.S., (2003) Thermal error measurement and modelling in machine tools. Part II. Hybrid Bayesian Network—support vector machine model, International Journal of Machine Tools & Manufacture.

Stander, C.J. and Heyns, P.S., (2001) Fault detection on gearboxes operating under fluctuating load conditions, Proceeding of the 14thInternational Congress on Condition Monitoring and Diagnostic Engineering Management Manchester UK 4-6 September, pp 457-464.

Stander, C.J. and Heyns, P.S., (2002)[a] Instantaneous Shaft Speed monitoring of gearboxes under fluctuating load conditions, Proceeding of the 15th International Congress on Condition Monitoring and Diagnostic Engineering Management, Birmingham UK 2-4 September 2002, pp 220-230.

Stander, C.J., Heyns, P.S. and Schoombie, W., (2002)[b] Using vibration monitoring for local fault detection on gears operating under fluctuating load conditions, Mechanical Systems and Signal Processing 16(6), pp 1005-1024.

Stander, C.J. and Heyns, P.S., (2003) Condition monitoring of gearboxes under cyclic and non-cyclic loading conditions, Proceeding of the 16th International Congress on Condition Monitoring and Diagnostic Engineering Management, Sweden 27-29 August, pp 601-610.

Staszewski, W.J. and Tomlinson, G.R., (1994) Application of the Wavelet Transform to fault detection in a spur gear, Mechanical Systems and Signal Processing 8(3), pp 289-307.

Staszewski, W.J., Worden, K. and Tomlinson, G.R., (1997) Time-Frequency analysis in gear fault detection using the Wigner-Ville and Pattern recognition, Mechanical Systems and Signal Processing 11(5), pp 673-692.

Stewart, R.M., (1977) Some useful data analysis techniques for gear diagnostics, Institute of Sound and Vibration Research, Paper MHM/R/10/77.

Taurino A.M., Distante, C., Siciliano, P. and Vasanelli, L., (2003) Quantitative and qualitative analysis of VOCs mixtures by means of a micro sensors array and different evaluation methods, Sensors and Actuators B 93, pp 117-125.

Trimble C.R., (1968) What is signal averaging?, Hewlett-Packard Journal 19(8), pp. 2-7.

Vapnik, V., Golowich, S. and Smola, A., (1997) Support vector method for function approximation, regression estimation, and signal processing. In Mozer, M., Jordan, M. and Petsche, T., editors, Advances in Neural Information Processing Systems 9, pp 281-287, Cambridge, MIT Press.

Vapnik, V.N., (1995) The nature of statistical learning theory, Springer-Verlag, New York, USA.

Walde, J.F., Tappeiner, G., Tappeiner, U., Tasser, E. and Holub, H.W., (2003) Statistical aspects of multilayer perceptrons under data limitations, Computational Statistics & Data Analysis.

Wang, W.J. and McFadden, P.D, (1993) Early detection of gear failure by vibration analysis-II. Interpretation of the time-frequency distribution using image processing techniques, Mechanical Systems and Signal Processing 7(3), pp 205-215.

Wang, W.J. and McFadden, P.D., (1995) Application of Orthogonal Wavelets to early gear damage detection, Mechanical Systems and Signal Processing 9(5), pp 497-507.

Wang, W. and Wong, A.K., (2000) Linear prediction and gear fault diagnosis, Proceeding of the 13th International Congress on Condition Monitoring and Diagnostic Engineering Management Houston Texas 3-8 December, pp 707-807.

White, G., (1991) Amplitude demodulation- a new tool for Predictive maintenance, Sound and Vibration, pp 14-19.

Yang, H., Chan, L. and King, I., (2002) Support Vector Machine Regression for volatile stock market prediction, IDEAL 2002, LNCS 2412, pp 391-396.

Zacksenhouse, M., Braun, S., Feldman, M. and Sidahmed, M., (2000) Towards helicopter diagnostics from a small number of examples, Mechanical Systems and Signal Processing 14(5), pp 523-543.

Zhong, M., Ding, S.X., Lam, J. and Wang, H., (2003) An LMI approach to design robust fault detection filter for uncertain LTI systems, Automatica (39), pp 543-550.

Appendix A

A.1 Experimental set-up

A schematic diagram of the accelerated gear life test rig used in this work is presented in Figure A.1. Figure A.2 shows a diagram of the accelerated gear life test rig.

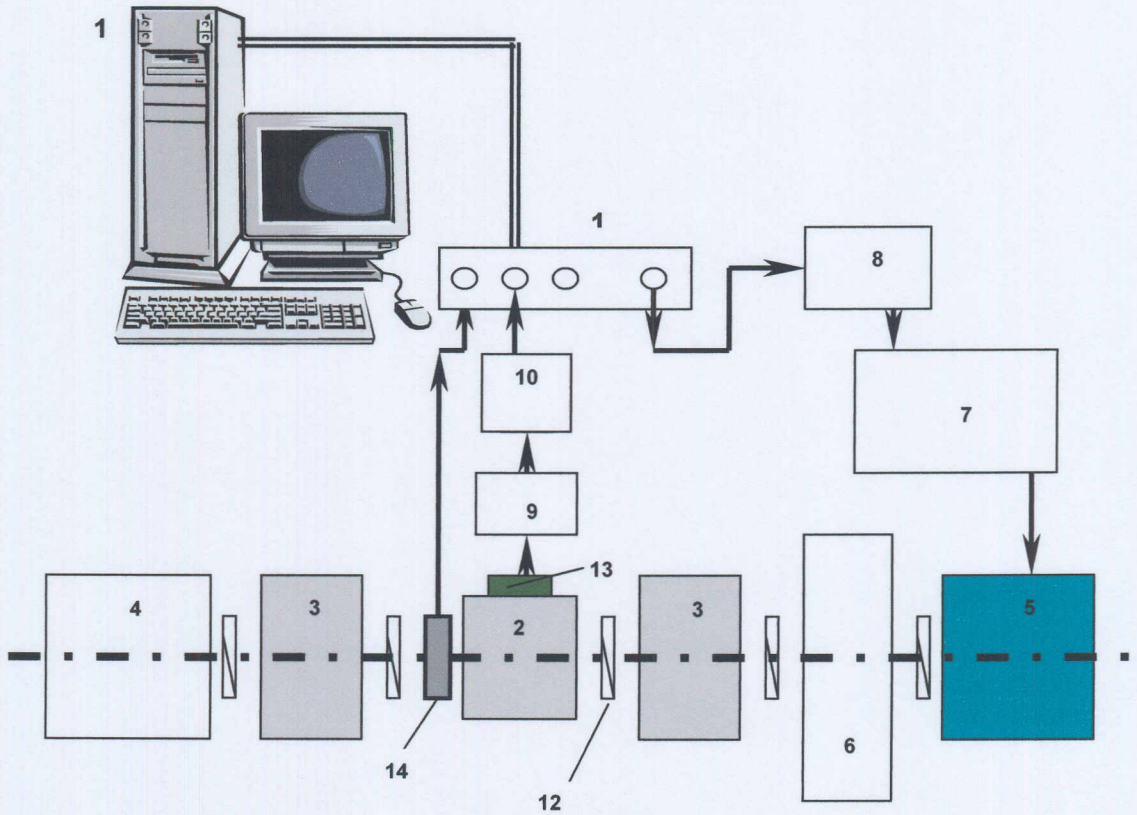


Figure A.1 Schematic diagram of experimental set-up

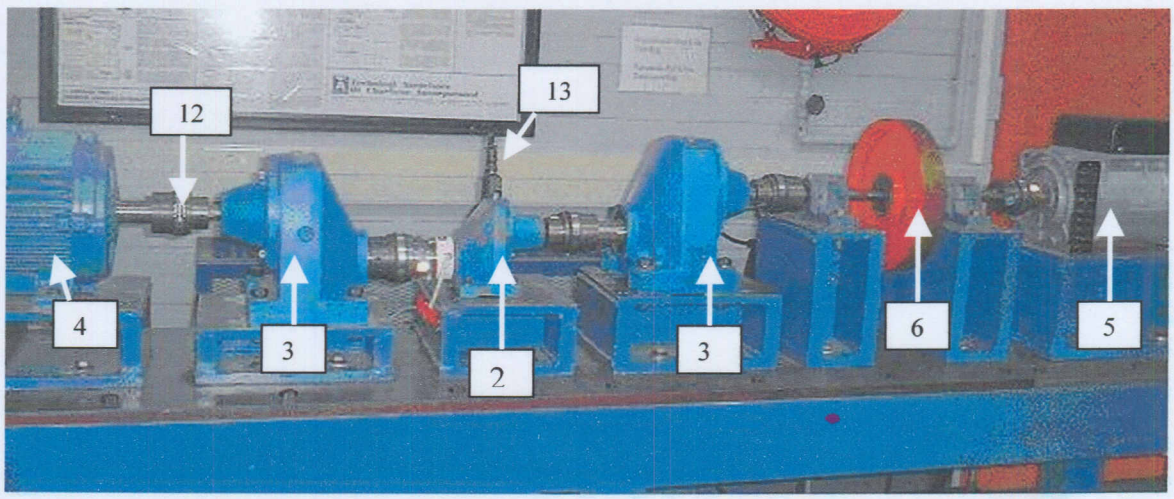


Figure A.2 Accelerated gear life test rig.

Table A1 give the specifications of the accelerated gear life test rig.

Table A.1 Accelerated life gear test rig Specifications

Item Number	Item	Description
1	PC	
2	Gearbox	Flender Himel Type E20A Ratio 1:96:1
3	Gearbox	Flender Himel Type E60A Ratio 4:72:1
4	Motor	WEG 380V / 50 Hz, three-phase
5	Alternator	Mecc alte 5.5 kVA, three-phase
6	Flywheel	Fenner 2517-25
7	Current controller	JEC current controller
8	DC Power supply	0-5V
9	Anti-aliasing low pass filter	4^{th} order low pass Butterworth filter with 300Hz cut-off
10	Signal conditioner	PCB ICP Model 482A22
11	Siglab analyser	Siglab model 20-42
12	Flexible couplings	
13	PCB accelerometer	5 V/g
14	Shaft encoder	Hengstler Himel type 0053 163 /10-30V DC /30mA

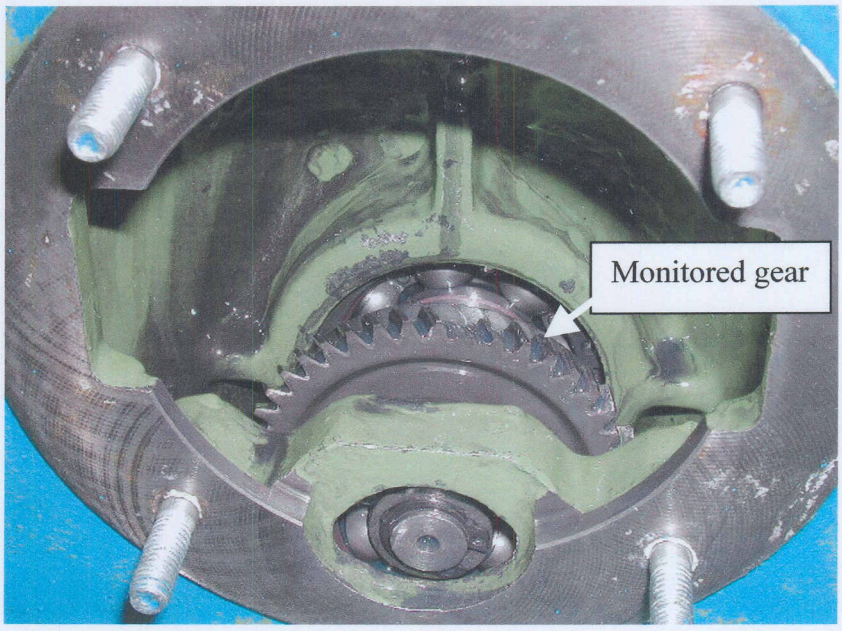


Figure A.3 Monitored gear inside the gearbox.

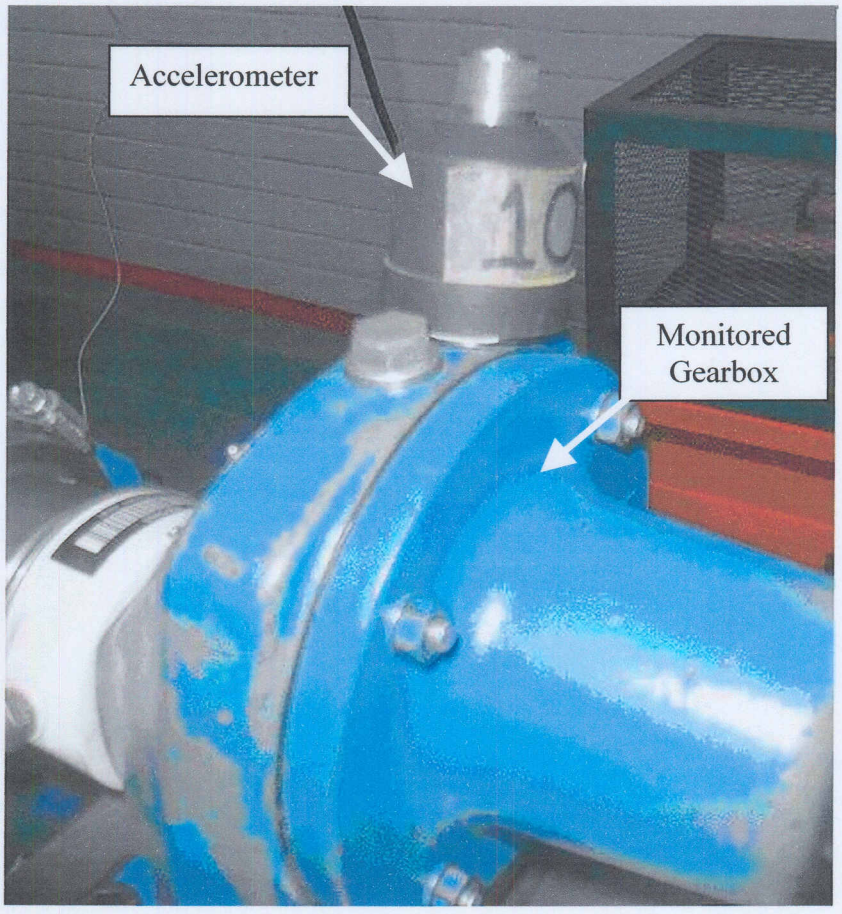


Figure A.4 Measurement point and mounting of accelerometer .

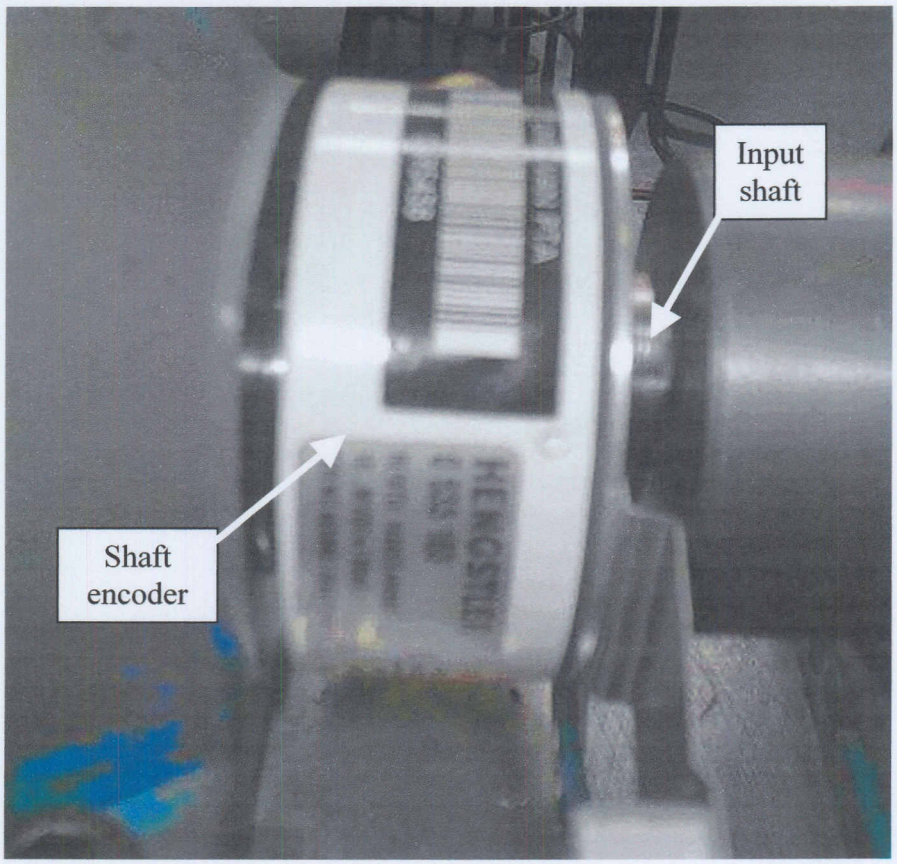


Figure A.5 Shaft encoder mounted on input to shaft to the monitored gearbox.

Appendix B

B.1 Back-Propagation method

In this study, the output is the time domain average of the rotation synchronised gearbox vibration data. In Figure 3.1 the output of the j^{th} hidden unit is obtained by calculating the weighted linear combination of the d input values.

$$a_j = \sum_{i=1}^{d} W_{ji}^{(1)} X_i \tag{B.1}$$

Where $W_{ji}^{(1)}$ indicates weights in the first layer, going from input i to hidden unit j while $W_{j0}^{(1)}$ indicates the bias for the j^{th} hidden unit. The activation of the j^{th} hidden unit is obtained by transforming the output a_j in equation (B.1) into z_j, which is shown in Figure 3.1, is

$$z_j = f_{inner}(a_j) \tag{B.2}$$

The output of the second layer is obtained by transforming the activation of the second hidden layer using the second layer weights. Given the output of the hidden layer z_j in equation (B.2), the output of unit k is given by

$$a_k = \sum_{j=0}^{M} W_{kj}^{(2)} y_j \tag{B.3}$$

Similarly equation (B.3) may be transformed into the output units by using some activation function as follows:

$$y_k = f_{outer}(a_k) \tag{B.4}$$

Combining equations (B.1), (B.2), (B.3) and (B.4) the input x to the output y can be related by a two-layered non-linear mathematical expression, which may be written as follows:

$$y_k = f_{outer}\left(\sum_{j=0}^{M} w_{kj}^{(2)} f_{inner}\left(\sum_{i=0}^{d} w_{ji}^{(1)}\right) + w_{k0}^{(2)}\right) \tag{B.5}$$

Where d is the number of input units, M is the number of hidden units, w_{ij} is the weight-vector, the function f_{outer} is linear and f_{inner} is a hyperbolic tangent function. These functions are defined as:

$$f_{outer}(v) = v \tag{B.6}$$

and

$$f_{inner}(v) = \tanh(v) = \frac{e^{v} - e^{-v}}{e^{v} + e^{-v}} \tag{B.7}$$

The weights w_i and in the hidden layers are varied until the error between the network prediction and the output from the training data is minimised.

Given the training set $D = \{X_k, t_k\}_{k=1}^{N}$ and assuming that the targets t_k are sampled independently given the inputs x_k and the weight parameters w_{kj} the sum of square of error cost function E is given by

$$E = \frac{1}{2}\sum_{n}\sum_{k}(y_{nk} - t_{nk})^2 \tag{B.8}$$

Where n is the index for the training pattern and k is the index for the output units.

The minimisation of E is achieved by solving for the derivative of the error in equations (B.8) with respect to the weights. The derivative of the error is calculated with respect to the weights that connects the hidden layer to the output layer and may be written using the chain rule as follows:

$$\begin{aligned}\frac{\partial E}{\partial w_{kj}} &= \frac{\partial E}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} \\ &= \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} \\ &= \sum_{n} f'_{outer}(a_k)\frac{\partial E}{\partial y_{nk}} z_j\end{aligned} \tag{B.9}$$

where z_j is given in equation (B.2). The derivative of the error with respect to the weights which connects the hidden layer to the output layer may be written using the chain rule is given by

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}}$$
$$\frac{\partial E}{\partial w_{kj}} = \sum_n f'_{inner}(a_j)\sum_k w_{kj} f'_{outer}(a_k)\frac{\partial E}{\partial y_{nk}} \tag{B.10}$$

The derivative of the sum of square cost function in equation (B.8) is written as

$$\frac{\partial E}{\partial y_{nk}} = t_{nk} - y_{nk} \tag{B.11}$$

The derivatives of the linear activation function in equation (B.6) is:

$$f'_{outer}(a_k) = c \tag{B.12}$$

while the derivative of the hyperbolic tangent function is:

$$f'_{inner}(a_j) = \text{sech}^2(a_j) \tag{B.13}$$

This appendix shows the derivatives of the errors with respect to weights. Equation (B.11) shows the derivative of the cost functions that could be incorporated into equations (B.9) and (B.10). Equations (B.12) and (B.13) show the derivatives of the two possible activation functions.

Appendix C

C.1 Gradient method

In this appendix the scaled conjugate optimisation method is described. Before introducing the scaled conjugate gradient method the conjugate gradient method is introduced. In supervised neural network training, the main goal is to identify weights that give the best prediction of the output whenever presented with the input. The scaled conjugate gradient method is used to sample through the weight space until the weight vector that minimises the distance between the neural network prediction and the target data is obtained.

C.2 Conjugate gradient method

The weight vector that gives the minimum error is obtained by taking successive steps through the weight space as follows:

$$w(n+1) = w(n) + \Delta w(n) \tag{C.1}$$

where n is the iteration step and Δ represents change. Different algorithms choose this step size differently. In this section, gradient descent method will be discussed, followed by how it is extended to the conjugate gradient method. For the gradient descent method, the step size in equation (C.1) is defined as:

$$\Delta w^n = -\eta \nabla E(w(n)) \tag{C.2}$$

where the parameter η is the learning rate and the gradient of the error is calculated using the back-propagation technique described in Appendix B. If the learning rate is sufficiently small, the value of error will decrease at each successive step until a minimum is obtained. The disadvantage with this approach is that it is computationally expensive compared to other techniques.

For the conjugate gradient method the quadratic function of error is minimised at each iteration over a progressively expanding linear vector space that includes the global minimum of the error. For the conjugate gradient procedure, the following steps are followed (Haykin, 1999; Marwala, 2001):

- Choose the initial weight $w(0)$.
- Calculate the gradient vector $\nabla E(w(0))$.

At each step n use the line search to find $\eta(n)$ that minimises $E(\eta)$ representing the cost function expressed in terms of η for fixed values of w and $-\nabla E(w(0))$.

- Check that the Euclidean norm of the vector $-\nabla E(w(n))$ is sufficiently less than that of $-\nabla E(w(0))$.
- Update the weight vector $w(n+1) = w(n) - \eta(n)\nabla E(w(n))$. For $w(n+1)$ compute the updated gradient $\nabla E(w(n+1))$.
- Use Polak-Ribiére method to calculate $\beta(n+1)$

$$\beta(n+1) = \frac{\nabla E(w(n+1))^T (\nabla E(w(n+1)) - \nabla E(w(N)))}{\nabla E(w(n))^T \nabla E(w(n))} \tag{C.3}$$

- Update the direction vector $\nabla E(w(n+1)) = \nabla E(w(n+1)) - \beta(n+1)\nabla E(w(n))$.
- Set $n = (n+1)$ and go back to step 3.
- Stop when the condition $\|\nabla E(w(n))\| = \varepsilon \|\nabla E(w(0))\|$ is satisfied in which ε is a small number.

C.3 Scaled conjugate gradient method

The scaled conjugate gradient method differs from conjugate gradient method in that it does not involve the line search described in step 3 in the previous section.
The step-size (see step 3) is calculated by using the following formula (Møller, 1993):

$$\bar{\eta}(n) = 2\left(\eta(n) - \frac{\nabla E(n)^T H(n)\nabla E + \eta(n)\|\nabla E(n)\|^2}{\|\nabla E(n)\|^2}\right) \tag{C.4}$$

where H is the Hessian matrix of the gradient.

Appendix D

D.1 Feature space

This appendix discusses the methods that can be used to construct a mapping into a high dimensional feature space by the use of reproducing kernels and implementation issues with regards to SVM. The idea of kernel functions is to enable operations to be performed in the input space rather than a potentially higher dimensional feature space. This provides a way of addressing the curse of dimensionality. The computation is however still critically dependent upon the number of training patterns and to provide a good data distribution for the high dimensional problem will generally require a large training set.

D.1 Kernel functions

The following theory is based on the Reproducing Kernel Hilbert Space (RKHS) (Aronszajn, 1950; Gunn , 1998). An inner product in the feature space has an equivalent kernel in input space,

$$K(x,x') = \langle \phi(x), \phi(x') \rangle, \tag{D.1}$$

provided certain conditions hold. If K is a symmetric positive definite function, which satisfies Mercer's condition,

$$K(x,x') = \sum_{m}^{\infty} a_m \phi_m(x) \phi_m(x'), \quad a_m \geq 0, \tag{D.2}$$

$$\iint K(x,x') g(x) g(x') dx dx' > 0, \quad g \in L_2, \tag{D.3}$$

Then the kernel represents a legitimate product in feature space. Valid functions satisfying the Mercer's conditions that were investigated in this study are given below. These functions are valid for all real x and x' unless otherwise stated.

D.2.1 Gaussian radial basis function

Radial basis functions have received significant attention, most commonly with a Gaussian of the form,

$$K(x,x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right). \tag{D.4}$$

Classical techniques utilising radial basis functions employ some method of determining a subset of centres. Typically the method of clustering is employed to select the subset of centres. An attractive feature of the SVN is that this selection is implicit, with each support vector contributing one local Gaussian function, centred at that data point. By further consideration it is possible to select the global basis function width, s, using the SRM principle (Vapnik, 1995).

D.2.2 Exponential radial basis function

The form below defines an exponential radial basis function,

$$K(x,x') = \exp\left(-\frac{\|x-x'\|}{2\sigma^2}\right). \tag{D.5}$$

This form produces a piecewise solution that can be attractive when discontinuities are acceptable.

D.2.3 Splines

Splines are a popular choice for modelling because of their flexibility. A finite spline, of order κ, with N knots located at τ_s is given by,

$$K(x,x') = \sum_{r=0}^{\kappa} x^r x'^r + \sum_{s=1}^{N} (x-\tau_s)_+^{\kappa} (x'-\tau_s)_+^{\kappa}. \tag{D.6}$$

An infinite spline is defined on the interval [0, 1) by

$$K(x,x') = \sum_{r=0}^{\kappa} x^r x'^r + \int_0^1 (x-\tau_s)_+^{\kappa} (x'-\tau_s)_+^{\kappa} \, d\tau. \tag{D.7}$$

In the case where $\kappa = 1, \left(S_1^\infty\right)$, the kernel is defined by,

$$K(x,x') = 1 + \langle x,x' \rangle + \frac{1}{2}\langle x,x' \rangle \min(x,x') - \frac{1}{6}\min(x,x')^3, \tag{D.8}$$

where the solution is piecewise cubic.

D.2.4 B-splines

Bsplines are another popular spline formulation. The B-splines kernel is defined on the interval [-1, 1], by the attractive closed form

$$K(x,x') = B_{2N+1}(x - x'). \tag{D.9}$$

D.3 Loss functions

Using the quadratic loss function in Figure 3.6 (a),

$$L_{quad}(f(\mathrm{x}) - y) = (f(\mathrm{x}) - y)^2. \tag{D.10}$$

The solution is given by,

$$\begin{aligned} \max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} & -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle x_i, x_j \rangle \\ & + \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)y_i - \frac{1}{2C}\sum_{i=1}^{l}\left(\alpha_i^2 + (\alpha_i^*)^2\right). \end{aligned} \tag{D.11}$$

The corresponding optimisation can be simplified by exploiting the KKT conditions, and noting that these imply $\beta_i^* = |\beta_i|$. The resultant optimisation problem is,

$$\min_{\beta} \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\beta_i\beta_j\langle x_i, x_j \rangle - \sum_{i=1}^{l}\beta_i y_i + \frac{1}{2C}\sum_{i=1}^{l}\beta_i^2 \tag{D.12}$$

with constraints,

$$\sum_{i=1}^{l} \beta_i = 0. \tag{D.13}$$

For the Huber loss function in Figure 3.6 (c),

$$L_{Hubber}\left(f(\mathrm{x})-y\right)=\begin{cases}\frac{1}{2}\left(f(\mathrm{x})-y\right)^2 & \text{for} \quad |f(\mathrm{x})-y|<\mu \\ \mu|f(\mathrm{x})-y|-\frac{\mu^2}{2} & \text{Otherwise}\end{cases}, \tag{D.14}$$

the solution is given by,

$$\begin{aligned}\max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} & -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle \\ & +\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i-\frac{1}{2C}\sum_{i=1}^{l}\left(\alpha_i^2+(\alpha_i^*)^2\right)\mu.\end{aligned} \tag{D.15}$$

The resultant optimisation is

$$\min_{\beta}\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\beta_i\beta_j\langle x_i,x_j\rangle-\sum_{i=1}^{l}\beta_i y_i+\frac{1}{2C}\sum_{i=1}^{l}\beta_i^2\mu \tag{D.16}$$

with constraints,

$$\begin{aligned}&-C\le\beta_i\le C, \qquad i=1,\mathrm{K}\ ,l \\ &\sum_{i=1}^{l}\beta_i=0.\end{aligned} \tag{D.17}$$

D.4 Implementation issues

For SVM the resulting optimisation problems are dependent upon the number of training examples. For large data sets methods have been proposed for speeding up the algorithm by decomposing the problem into smaller ones. The optimisation problem for an ϵ-insensitive loss function can be expressed in matrix format as,

$$\min_{x}\frac{1}{2}x^T H x+c^T x \tag{E.18}$$

where

$$H = \begin{bmatrix} XX^T & -XX^T \\ -XX^T & XX^T \end{bmatrix}, \quad c = \begin{bmatrix} \varepsilon + Y \\ \varepsilon - Y \end{bmatrix}, \quad x = \begin{bmatrix} \alpha \\ \alpha^* \end{bmatrix} \tag{E.19}$$

with constraints

$$x \cdot (1,\mathrm{K},1,-1,\mathrm{K},-1) = 0, \quad \alpha_i, \alpha_i^* \geq 0, \quad i = 1,\mathrm{K},l. \tag{E.20}$$

where

$$X = \begin{bmatrix} \mathbf{x}_1 \\ \mathrm{M} \\ \mathbf{x}_l \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ \mathrm{M} \\ y_l \end{bmatrix} \tag{E.21}$$