APPROVAL SHEET

Title of Thesis: Fault Detection Using an Artificial Neural Network and Variable Structure Based Approach

Name of Candidate: Andrew Lee
Master of Science, 2017

Thesis and Abstract Approved: ________________
(Dr. S. Andrew Gadsden)
(Assistant Professor)
(Mechanical Engineering)

Date Approved: July 6, 2017

ABSTRACT

Title of Document:	FAULT DETECTION USING AN ARTIFICIAL NEURAL NETWORK AND VARIABLE STRUCTURE BASED APPROACH
	Andrew Lee
Directed By:	Dr. S. Andrew Gadsden, Assistant Professor, Department of Mechanical Engineering

The ultimate goal of fault detection and isolation is to maximize the life span of equipment and minimize the cost of maintenance. The development of intelligent diagnostic, prognostic, and health management technology has proven to be important for industrial and defense maintenance procedures in recent years. While diagnostic technology for aircraft have existed for more than 50 years, modern CPUs permit on-board intelligent and estimation-based fault detection methods. This thesis discussed two strategies in particular: artificial neural networks and smooth variable structure filters. The purpose of this thesis is to propose a method of health state awareness for a helicopter blade using an artificial neural network as well as develop a variable structure-based fault detection and diagnosis strategy for an electromechanical actuator.

FAULT DETECTION USING AN ARTIFICIAL NEURAL NETWORK AND VARIABLE STRUCTURE BASED APPROACH

By

Andrew Lee

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County, in partial fulfillment
of the requirements for the degree of
Masters of Science in
Mechanical Engineering
2017

Acknowledgements

I express sincere gratitude to my advisor, Dr. S. Andrew Gadsden, for his continuous support and guidance throughout my graduate career.

Furthermore, I would like thank my committee members Dr. Anne Spence, Dr. Chuck Eggleton, and Dr. Marc Zupan. To Brian, Chris, Elyse, Eugene, Jinho, Kevin, Nate, Neal, Stephanie and all ASRL members, thank you for making these last 2 years so entertaining.

Last but not least, I would like to thank Dr. Drew Wilkerson for his support of ASRL's quadcopter, 3D printing, and laser cutting efforts.

Table of Contents

List of Figures

List of Equations

Chapter 1: Introduction

1.1 Overview

The development of intelligent diagnostic, prognostic, and health management technology has proven to be important for industrial and defense maintenance procedures in recent years [1]. Fault detection technology is evolving at a rapid pace along with expanding application domains and an increasing customer base [1]. However, the implementation of fault diagnosis, prognosis, and maintenance techniques can be difficult as it is often an interdisciplinary process involving mechanical, electrical, and computer engineering with in-depth knowledge on fault mechanisms and signal acquisition/processing techniques.

The ultimate goal of fault diagnosis and prognosis is to maximize the life-span of equipment and minimize the cost of maintenance [1]. With the increasing complexity and cost of machinery, there is a need for innovation in order to achieve an accurate fault diagnosis and prognosis. Many everyday systems are comprised of dense electronic circuitry and intricate mechanical interactions between components. A typical automobile consists of approximately 2,000 functional components, 30,000 parts, and 10 million lines of software code [2,3]. In the past, fault detection consisted mainly of planned maintenance schedules which were time consuming and labor intensive. Like many tasks, fault detection is now trending towards automation.

Diagnostic technology for aircraft have existed for more than 50 years in some capacity [1]. Improvements in diagnostic capability for aircraft have largely been due to advances in computing technology. Modern CPUs with small form factors allow for on-board processing of data which prevents data loss or incorrectly processed data at the ground station [1]. In addition, modern CPUs are powerful enough to process the large volume of data made available by Multiplex data buses on aircraft for fault detection [1]. Intelligent methods such as artificial neural networks (ANN) can be used to find relationships within rich data sets to infer upon the presence of fault precursors.

1.2 Research Objectives and Goals

There are many fault detection techniques with an even broader amount of applications and viable system environments. This thesis will discuss two strategies in particular: artificial neural networks and smooth variable structure filters (SVSF). **The goal of this thesis is to propose a method for health state awareness of a helicopter blade using an artificial neural network as well as develop a variable structure-based fault detection and diagnosis strategy for an electromechanical actuator [4, 5].**

The proposed experimental setup for a helicopter blade uses a single axis electrodynamic shaker to subject a scale aluminum helicopter blade to transverse vibratory excitation at the hub [4]. Data collected from an array of accelerometers, piezo electric transducers, and strain gauges embedded along the blade would be used to train a neural network for fault detection and diagnosis. The training phase would include healthy samples as well as samples with known fault types and locations. In order to test the efficacy of the proposed setup, a preliminary study was performed using an accelerometer adhered to an aluminum cantilever beam. Vibrational data was recorded after the cantilever beam was released from an initial deflection.

In addition to ANN, a variable structure based approach was used for fault detection in am electromechanical system. The Kalman filter (KF) is one of the most popular methods used for state estimation and signal processing [5]. However, this method requires strict assumptions in order to provide an optimal solution to the estimation problem [5]. The smooth variable structure filter (SVSF) provides a solution to overcome robustness issues. This thesis uses the SVSF on a dynamic model of an electrohydrostatic actuator (EHA) for fault detection and diagnosis.

1.3 Organization of Thesis

The thesis is organized into the following chapters. Chapter 2 contains a literature review of aerospace fault detection, ANN strategies, and estimation based techniques. Chapter 3 contains a proposed methodologies for using an ANN for health state awareness of a helicopter blade. The proposed setup is justified by a preliminary setup showing the efficacy of the ANN. Chapter 4 explores the application of a SVSF for fault detection of an electromechanical actuator. Finally, Chapter 5 contains concluding remarks of the presented work and possible improvements for future study.

Chapter 2: Literature Review

This chapter provides a summary of the main literature used to support research in this thesis. Section 2.1 provides background of fault detection in the automotive and aerospace sector. Section 2.2 highlights examples of ANNs in fault detection applications. Section 2.3 gives an overview of estimation-based strategies such as the Kalman filter (KF) and SVSF.

2.1 Fault Detection and Isolation

Fault detection and isolation (FDI) refers to the monitoring of a system in order to identify when a fault has occurred as well as ascertaining the location of the fault. There is a strong demand for reliability in and safety in automotive and aerospace engineering as automated systems become more prevalent. Artificial intelligence (AI) methods such as fuzzy logic and neural networks are widely used in FDI systems in an effort to provide improved reliability and a reduction in the probability of false positives.

There are three main categories of FDI techniques: signal-based fault detection, model-based fault detection, and artificial intelligent techniques [6]. Signal-based fault detection works by extracting by extracting the fault signature and comparing the systems against healthy operational trends [6]. Useful data is extracted in the time-domain, frequency-domain or time-frequency domain for extracting fault signatures [6, 7]. Common signals used in FDI techniques to detect faults include vibration signals, pressure signals, and noise levels. One of the most widely used feature extraction techniques is qualitative trend analysis (QTA) [6]. This is a data-driven FDI methodology in which features (trends) are extracted form measured signals in order to determine necessary countermeasures. **This methodology is explored further in Chapter 3**.

QTA has been applied extensively as a means of FDI [6, 8]. In addition, features can be extracted using discrete wavelet-based techniques (DWT) [6, 9]. Fault detection using DWT was broadly deliberated by Postalcioglu et al. [6, 10]. DWT can be divided into two major processes: measured signal decomposition, followed by signal edge detection that may occur due to faults [6]. Model-based FDI is mainly based on residual generation which represent inconsistencies between the actual physical system measurements and the mathematical model of the system [6]. Model-based FDI techniques can generate residuals by using parameter estimation, observers, and parity space comparison [6]. Fault detection strategies are widely used in the automotive and aerospace sectors.

2.1.1 Fault Detection in the Automotive Sector

Signal-based FDI techniques for internal combustion engine fault detection have been explored extensively [6, 11, 12]. FDI techniques frequently use noise levels, as well as pressure or vibration signals, in order to detect faults [6]. Chung et al. implemented an engine fault detection methodology using sound measurements by acquiring sound intensities using microphones in 1979 [6, 13, 14]. This particular method is one of the oldest techniques that were utilized at the General Motors research laboratories [6, 13]. The microphones were able to effectively generate a detailed mapping of engine noise using cross-spectral analysis [6, 13]. By using a noise source ranking methodology, this technique is able to identify the noise source [6, 13]. Leitzinger provided a comparison between accelerometers, microphones, and laser Doppler vibro-meters, to identify engine faults [6, 14, 15]. According to the research, microphones provide an easy, contactless measuring system. However, it is possible generate inconsistent results and produce false alarms [6, 15]. Furthermore, the study concluded that accelerometers and laser Doppler vibro-meters were able provide more reliable measurements [6, 15]. Acoustic tests on internal combustion engines in a production environment using two overhead microphones to measure sound pressure are described by Jonuscheit [6, 16]. Rizzoni and Min used a real time model-based technique in order to diagnose sensor failures in automotive engine control systems [6, 17]. Faults considered include the manifold absolute pressure and throttle position sensors and experimental results demonstrate the effectiveness of the proposed technique [6, 17].

2.1.2 Fault Detection in the Aerospace Sector

There has been steady progression in aerospace fault diagnosis and prognosis. Implementations have evolved from manual to semi-automated and fully automated methods. Early generations of aircraft depended on manual fault detection and health management on the ground [1, 18]. The aircraft systems were mostly analog and independent from one another. These systems required voltmeters and schematics in order to troubleshoot their problems [1, 19].

In the 1950s and 1960s, aircraft systems eventually incorporated built-in test equipment (BITE) such as alarms and simple trending analysis in order to notify operators of critical failures [1, 20]. Eventually, original equipment manufacturers started placing fault indicators directly on line-replaceable units. Many line-replaceable units contained a self-test switch with LED fault indicators. BITE replaced many ground support testing procedures and in some cases the BITE showed the location of the fault as well [1].

When computer systems were integrated with BITE in the 1970s they became known as *digital* BITE [1, 19]. Numeric codes were displayed on the front panel of the line-replaceable units that corresponded to specific faults that could be looked up in a manual [1, 18]. *Digital* BITES were employed by many planes such as the Boeing 757/767 and the Airbus A300/310 [1].

As aircraft technology became more complex, the number of systems that needed to be monitored increased. There were numerous line-replaceable units in a single aircraft, each with its own unique fault detection display method. Monitoring all the separate units was cumbersome. Thus, a centralized system became necessary. Digital data buses like the ARINC 429 (Mark 33 Digitial Information Transfer System) allowed for autopilot systems to communicate with sensor data from other subsystems [1, 20]. The Boeing 767 had a maintenance control panel that enabled automatic landing by combining the maintenance functions of related systems [1].

In 1986, the ARINC 604 established a central fault display system (CFDS) which aggregated all maintenance indication into a single interface [1, 21]. Figure 1 shows the network between the central fault detection interface unit (CFDIU) and other systems. This streamlined the fault monitoring process and eliminated the need for front panel displays on individual line-replaceable units. This system was incorporated into several aircraft including the Boeing 737 and the Airbus A320/330/340 [1].

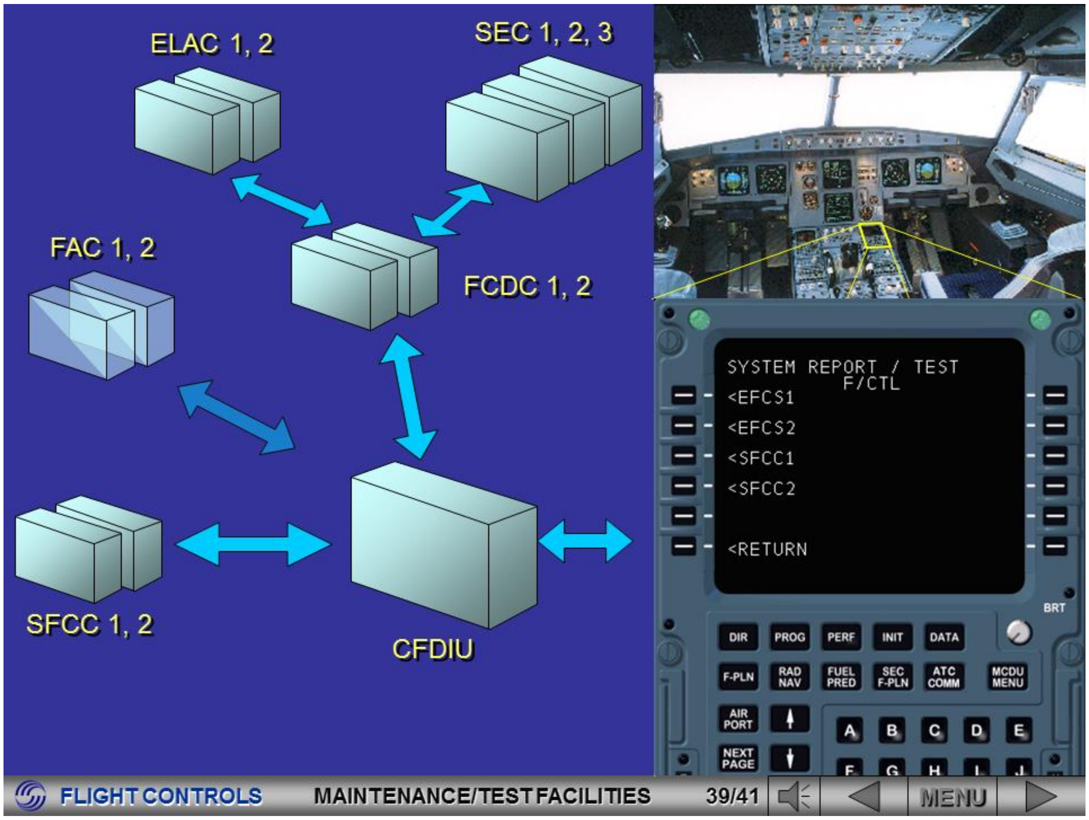


Figure 1: The FAC sends BITE data to the CFDIU to be displayed: tests can then be triggered accordingly [21].

As systems became increasingly complex, faults in one system could trigger fault indications in several other systems. Thus, it was often difficult for mechanics to identify the true source of the fault. This issue was mitigated the ARINC 624 in the 1990s [1]. This new standard consolidated fault indication from multiple subsystems and added additional functionality to support condition-based maintenance [1]. This reduced the need for ground testing even further.

There are many instances in which the specific unit that is the source of the fault can be identified through the CFDS. However, there are equally as many situations such as structural health monitoring where it is not possible without the aid of additional sensors or wiring. This particular application is important because failure of a wing or helicopter blade can be catastrophic.

Structural health monitoring involves mounting or imbedding sensors into a structure in order to detect faults and fatigue earlier than manual inspection [22]. By creating a network of sensors mounted on an aircraft, an operator would be able to monitor the structural health constantly in real time, eliminating the need for periodic checkups [22]. One of the sensors of interest in structure health monitoring is the piezoelectric transducer. Piezoelectric materials such as Lead Zirocondate Titanate (PZT) generate an electrical charge under mechanical stress. Conversely, the material will experience a geometric change when a voltage is applied. This property makes piezoelectric materials suitable as both actuators and sensors for vibrational data acquisition. Figure 2 depicts researchers mounting piezo electric sensors on a printer circuit board on an aircraft structure [23].

Figure 2: Dennis Roach and Ciji Nelson mounting piezoelectric sensor to a commuter jet at Sandia National Laboratory [23].

In addition to piezoelectric transducers, accelerometers and strain gauges could provide additional information about the structural health of an aircraft. Useful information can be extracted from the electrical data such as natural frequency and vibrational amplitudes and fed into an artificial neural network.

2.2 Artificial Neural Networks

ANNs are proven to be an effective tool for FDI [6,24]. This is attributed to several characteristics of ANNs: they have powerful self-learning and self-adapting characteristics, effective online adaptation algorithms (besides their parallel and pipeline processing characteristics), good noise rejection capabilities, and excellent nonlinear approximation properties [6, 25]. In addition, ANNs provide the ability to include models with only partly known physical structure, resulting in semi-physical models [6, 26].

Since ANN-based fault detection technique represents a black-box approach applied to any fault condition without the need to know the specific crack angle where the fault occurs or a specific frequency, ANN-based fault detection has a competitive advantage over other FDI techniques such as Wavelet analysis [6, 14]. Thus, additional fault conditions can be further added to the algorithm. Finally, as more data sets become available, the ANN learning capability increases significantly and potentially produce higher classification accuracy [6].

2.2.1 Fault Detection Using ANN Techniques

Yedavalli details a neural network-based adaptive observer for aircraft engine parameter estimation [6, 27]. This adaptive observer combines the KF with ANNs and is able to compensate for nonlinearities that cannot be handled simply by the filter [6]. Patton and Chen introduced a new approach to the design of optimal observer-based residual generators for detecting incipient faults in flight control [6, 28]. This method reduced the risk of generating false positives of fault detection [28]. In addition, an observer-based fault detection system in robots using nonlinear and fuzzy logic residual evaluation is discussed by Schneider and Frank [6, 29]. Yedavalli and Rama presented a fault diagnostic scheme for aircraft engine sensor fault [6, 30]. The proposed algorithm can distinguish between modeling uncertainties and occurrence of faults in order to reduce false positives [6]. Vinsonneau et al. presented an observer-based FDI for the drive-train of a Jaguar vehicle involving an automatic transmission [31]. A model representing the drive-train of the vehicle is derived using nonlinear polynomials that relate manifold pressure, engine speed, and the wheel speed [6].

2.2.2 Multi-Layered Neural Networks

A multi-layered neural network (MNN) as shown in Figure 3, usually consists of an input layer of a set of sensory units or source nodes, L-1 hidden layers of neurons, and an output layer of neurons [32]. The input signal propagates through the network on a layer-to-layer basis in the forward direction. Each neuron actuates a response using sigmoidal activation function [32]. MNNs have been successfully applied in solving some difficult problems by training them in a supervised manner with the error back-propagation algorithm [32].

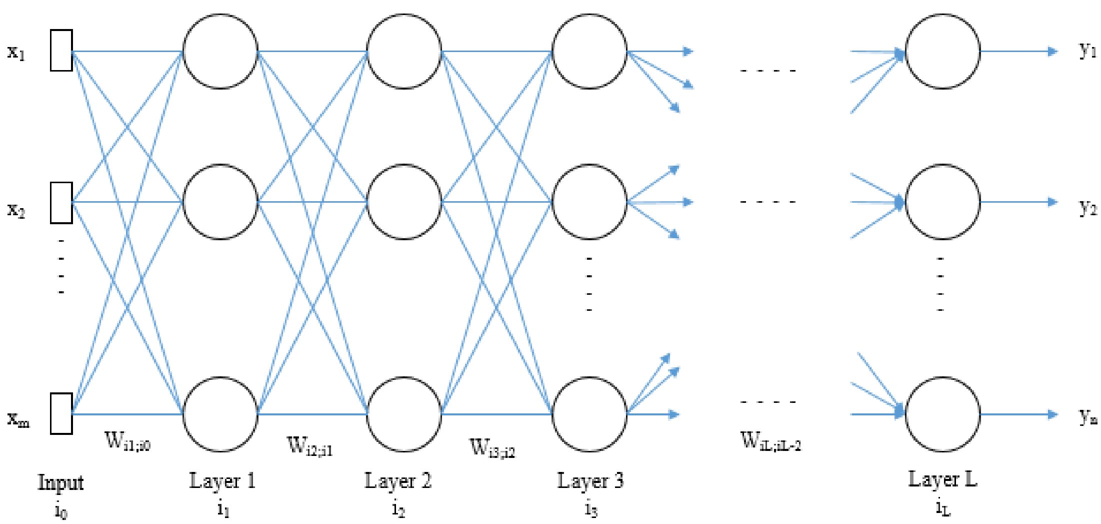


Figure 3: An L-layered neural network [32]

The synopsis of the notations used to represent an MLN with L layers is as follows:

x	m-by-1 input vector
y	n-by-1 output vector
i_k	$k = 1, \dots., L$: index for representing a neuron in the *k*th layer
i_0	index for representing a neuron in the input layer
$\mathrm{W}_{\mathrm{i}_k \mathrm{i}_{k-1}}$	Weight connecting i_k^{th} neuron of the *k*th layer and i_{k-1}^{th} of the *k*-1th layer

2.3 Estimation-Based Strategies

The process of extracting the value of a hidden quantity from indirect, inaccurate, or uncertain measurements is known as estimation. The hidden quantity may be a parameter or a state variable. The term *parameter* refers to a time-invariant physical quantity that may be a scalar, a vector, or a matrix. Although the term *time-varying parameter* may be appear in some texts, its variations must be slow in comparison to changes in the state variable [33]. The term *state* refers to a vector that evolves over time by the use of an equation which describes the dynamics of a system [33, 34]. The two different classes of estimators include the parameter estimator and the state estimator. The main goal of the estimation task is to minimize the state or parameter estimation error while being robust to uncertainties, noise, and perturbations. Noise and perturbations are inherently present in the measurement process, and are caused by instrument and environmental factors. System uncertainties are usually caused by inaccuracies in modeling the process, approximations, nonlinearities, and variations in physical parameters of the system [33].

Beginning in the fifteenth century, major contributions to the estimation field were made by a large number of contributors from a variety of backgrounds. The first major contributor to this field, Thomas Bayes, introduced the famous Bayesian rule for statistical inference that provides the basic formula for Bayesian estimation methods [33, 35]. Carl Friedrich Gauss pioneered the study that provides an optimal estimate from noisy data [33]. He invented the famous least square estimation method in 1795 and used it to solve nonlinear estimation problems in mathematical astronomy [33, 36]. Later on, Andrei Markov introduced the Markov process and Markov chain theories based on probability and statistical methods [33, 37]. The Markov theories formulate transitions in random processes from one state to another, between a finite or countable number of possible states [33]. He proved that the probability distribution of states may be calculated using its current distribution that contains the effects of all the past events of the system [33, 34]. In 1933, Andrei Kolmogorov laid down the modern axiomatic foundations of probability theory. Sydney Chapman continued the research on the Markov processes. Chapman and Kolmogorov independently presented the Chapman-Kolmogorov equations used for solving basic equations in estimation theory [33, 37].

Ronald Aylmer Fisher is known for the Fisher information matrix which represents a measure of the amount of information extracted from a sample of values with a given probability distribution [33, 37]. In 1949, Norbert Wiener introduced the Wiener filter formulation for signal processing applications [33]. This filter reduces the amount of noise present in a signal in comparison with an estimation of the desired noiseless signal [33, 38]. Kolmogorov along with Wiener, made the foundation of estimation theories that were used later to develop the theory of prediction, filtering, and smoothing [33]. His research ultimately led to the derivation of an optimal estimator, which was formulated for continuous-time systems [33, 39] Meanwhile, Kolmogorov independently derived an optimal linear predictor for discrete-time systems [33, 40]. Their research would later become famous, known as the Wiener-Kolmogorov filter (WF) which is a predecessor to the Kalman filter [33, 35].

2.3.1 Kalman Filter

In 1960, Rudolf Kalman, building on the work his predecessors, introduced a new approach to linear filtering and prediction problems which later became known as the Kalman filter [33, 34]. The Kalman filter was successfully applied by NASA for the Apollo's guidance and navigation system and quickly became popular as the most practical method for state estimation [33, 34, 41]. The Kalman filter (KF) uses a linear dynamic model and sequential measurements of the system to provide an optimal state estimate in the presence of Gaussian noise (which is white with a mean of zero). A continuous version of the KF was later developed by Kalman and Bucy which later became known as the Kalman-Bucy filter [33, 42]. Some extensions to the KF formulation, such as linearization and approximation, led to the extended Kalman filter (EKF) and the unscented Kalman filter (UKF), respectively. These extensions allowed the KF to be implemented on nonlinear systems for the purpose of state and parameter estimation. Other advanced variants of the Kalman filter include the quadrature Kalman filter (QKF) [33, 43, 44], mixture Kalman filter (MKF) [33,46], and the cubature Kalman filter (CKF) [47].

2.3.2 Kalman Filter [KF] Equations

The KF has been broadly applied to problems covering state and parameter estimation, signal processing, target tracking, fault detection and diagnosis, and even financial analysis [48, 49, 50]. The success of the KF comes from the optimality of the Kalman gain in minimizing the trace of the a *posteriori* state error covariance matrix [48, 51, 52]. The trace is taken because it represents the state error vector in the estimation process [48, 53]. The following five equations form the core of the KF algorithm, and are used in an iterative fashion. Equations (1.1) and (1.2) define the a priori state estimate $\hat{x}_{k+1|k}$ and the corresponding state error covariance matrix $P_{k+1|k}$, respectively.

Equation 1.1-1.5: Kalman Filter

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k \tag{1.1}$$

$$P_{k+1|k} = AP_{k|k}A^T + Q_k \tag{1.2}$$

The Kalman gain K_{k+1} (1.3) is used to update the state estimate $\hat{x}_{k+1|k+1}$ as per (1.4). The gain makes use of an innovation covariance S_{k+1}, which is defined as the inverse term found in (1.3).

$$K_{k+1} = P_{k+1|k}C^T\left[CP_{k+1|k}C^T + R_{k+1}\right]^{-1} \qquad 1.3)$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left[z_{k+1} - C\hat{x}_{k+1|k}\right] \tag{1.4}$$

The a *posteriori* state error covariance matrix $P_{k+1|k+1}$ is then calculated (1.5), and is used iteratively, as per (1.2).

$$P_{k+1|k+1} = [I - K_{k+1}C]P_{k+1|k}[I - K_{k+1}C]^T + K_{k+1}R_{k+1}K_{k+1}^T \tag{1.5}$$

A number of different methods have extended the classical KF to nonlinear systems, with the most popular and simplest method being the extended Kalman filter (EKF) [48, 54, 55]. The EKF is conceptually similar to the KF; however, the nonlinear system is linearized according to its Jacobian. This linearization process introduces uncertainties that can cause numerical instability and inaccurate estimates [48, 55].

2.3.3 Smooth Variable Structure Filter [SVSF]

When given an upper bound on the level of unmodeled dynamics and noise, the SVSF is stable and robust to modeling uncertainties, noise, and disturbances [48, 56, 57, 58]. The SVSF method is model-based and may be applied to differentiable linear or nonlinear dynamic equations [48, 59, 60]. The basic estimation concept of the SVSF is shown in Figure 4.

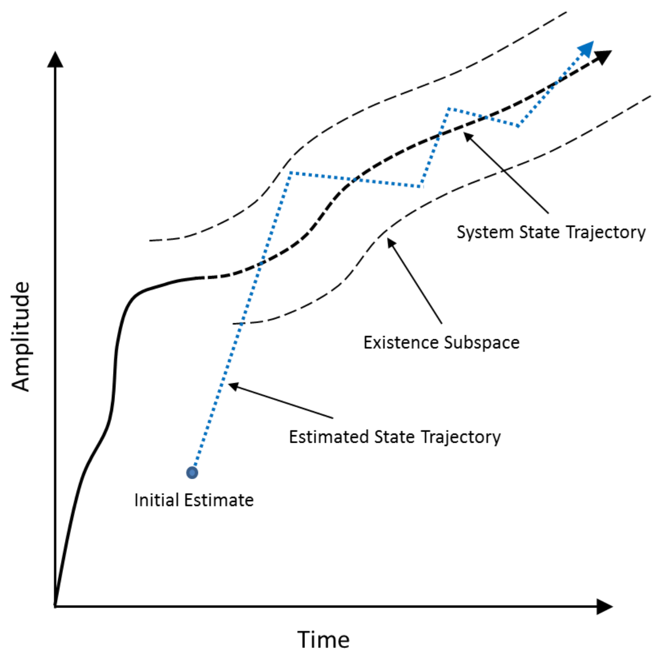


Figure 4: Standard SVSF estimation concept [51]

The SVSF estimation process is similar to the KF, with the main exception being the gain calculation [48, 61]. The predicted state estimates $\hat{x}_{k+1|k}$ and state error covariance $P_{k+1|k}$ are first calculated as per (1.1) and (1.2). Utilizing the predicted state estimates $\hat{x}_{k+1|k}$, the corresponding predicted measurements $\hat{z}_{k+1|k}$ and measurement errors $e_{z,k+1|k}$ may be calculated:

Equation 2.1-2.7: Smooth Variable Structure Filter

$$\hat{z}_{k+1|k} = C\hat{x}_{k+1|k} \tag{2.1}$$

$$e_{z,k+1|k} = z_{k+1} - \hat{z}_{k+1|k} \tag{2.2}$$

The SVSF gain is a function of: the *a priori* and the a *posteriori* measurement errors $e_{z,k+1|k}$ and $e_{z,k|k}$; the smoothing boundary layer widths ψ; and the 'SVSF' memory or convergence rate γ. The SVSF gain K_{k+1} is defined as follows [48, 51, 62]:

$$K_{k+1} = C_k^{+} diag\left[\left(\left|e_{z_{k+1|k}}\right| + \gamma\left|e_{z_{k|k}}\right|\right) \circ sat\left(\bar{\psi}^{-1} e_{z_{k+1|k}}\right)\right] diag\left(e_{z_{k+1|k}}\right)^{-1} \tag{2.3}$$

where $\circ$ signifies Schur (or element-by-element) multiplication and the superscript $+$ refers to the pseudoinverse of a matrix. The saturation function of (2.3) is defined by the following:

$$sat\left(\bar{\psi}^{-1} e_{z_{k+1|k}}\right) = \begin{cases} 1, & e_{z_i,k+1|k}/\psi_i \geq 1 \\ e_{z_i,k+1|k}/\psi_i, & -1 < e_{z_i,k+1|k}/\psi_i < 1 \\ -1, & e_{z_i,k+1|k}/\psi_i \leq -1 \end{cases} \tag{2.4}$$

where $\bar{\psi}^{-1}$ is a diagonal matrix constructed from the elements of the smoothing boundary layer vector ψ:

$$\bar{\psi}^{-1} = \begin{bmatrix} \frac{1}{\psi_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{\psi_m} \end{bmatrix} \tag{2.5}$$

The state estimates $\hat{x}_{k+1|k}$ and state error covariance matrix $P_{k+1|k}$ are updated respectively as per (1.4) and (1.5). Finally, the updated measurement estimate $\hat{z}_{k+1|k+1}$ and measurement errors $e_{z,k+1|k+1}$ are calculated, and are used in later iterations:

$$\hat{z}_{k+1|k+1} = C\hat{x}_{k+1|k+1} \tag{2.6}$$

$$e_{z,k+1|k+1} = z_{k+1} - \hat{z}_{k+1|k+1} \tag{2.7}$$

The existence subspace shown in Figure 4 represents the amount of uncertainties present in the estimation process, in terms of modeling errors or the presence of noise. The width of the existence space β is a function of the uncertain dynamics associated with the inaccuracy of the internal model of the filter as well as the measurement model, and varies with time [48, 62]. Typically this value is not exactly known but an upper bound may be selected based on *a priori* knowledge. When the smoothing boundary layer is defined larger than the existence subspace boundary, the estimated state trajectory is smoothed. However, when the smoothing term is too small, chattering remains due to the uncertainties being underestimated.

2.4 Summary

This chapter presented intelligent (ANN) and estimation-based methods (KF, SVSF) used in to address issues in FDI in literature. The following chapters will expand upon these concepts. Chapter 3 will propose a methodology for using an ANN for structural health monitoring of helicopter blades. Chapter 4 develops a SVSF strategy for FDI applications in an electromechanical system.

Chapter 3: Proposed Health State Awareness of Helicopter Blades

The proposed experimental setup uses multiple sensors such as accelerometers, piezoelectric transducers, and strain gages in order to extract relevant data to be fed into an ANN for FDI applications. In order to test the efficacy of the ANN, a preliminary study was conducted using only an accelerometer and aluminum cantilever beam specimens. Section 3.1 discusses instances where catastrophic failure of helicopter blades have occurred during operation. Section 3.2 gives an overview a feed-forward multilayer neural network. Section 3.3 proposes a method for structural health awareness while Section 3.4 contains the results of the preliminary study.

3.1 Motivation for Research

There has been considerable progress in the area of fault detection and isolation resulting in several possible approaches [63, 64, 65, 66, 67]. In the case of helicopter blades, early fault detection is vital in the prevention of catastrophic and possibly fatal mechanical failures. Amura et al. carried out an investigation of a military helicopter crash that resulted in the fatality of the entire crew [63, 68]. While four main rotor blades were found close to the impact point, a fifth blade was found approximately 900 m away from the wreck leading Amura et al. to conclude that the cause of the crash had been the failure of this blade. The rotor blade consists of a long hollow 6061-T6 aluminum extrusion. Visual examination of the fracture surface of the aluminum extrusion indicated fatigue crack growth followed by ductile overload separation.

Examination by optical and electronic microscopy of the fatigue fracture revealed an abnormal incision that appeared to be the fracture origin site. Further examination through electronic microscopy with X-EDS analyzer showed evidence the crack origin was the result of an inappropriate tool used to remove pockets during maintenance activities [63, 68]. In this instance, a neural network strategy may have proven useful as part of an NDT (non-destructive testing) regimen for early fault detection. While neural networks can be used simply as a fault detection device the ultimate goal is to devise a robust fault tolerant system as part of a hierarchical control scheme.

3.2 Overview of ANN Strategy

In general, there are two methods of approaching the analytical fault detection problem: the model-based approach and the signal-based approach. In the model-based approach, the engineer has access to an analytical or knowledge based model of the system whose behavior is being monitored. These models usually consist of linear systems which are more easily characterized. The data based approach bypasses the step of obtaining a mathematical model and deals directly with the data. This method has greater viability due to the non-linearity of a helicopter blade rotor system. One particular application of a neural network is monitoring a specimen in order to detect major changes in the operating system. In this case, the neural network is trained on a well-behaving system in order to detect anomalies in the output of the system being monitored. This paper proposes training a neural network on a system with known faults in parallel with a healthy system known as supervised learning. This method presents a challenge, however, in that it requires a representative sample of all faults in a training set in order to work properly [63, 69]. Helicopter blades have a wide spectrum of faults with varying degrees of severity. Thus, adequate training may prove resource and time intensive.

Neural networks are typically implemented in applications such as automatic vehicle control [63, 70], pattern recognition [63, 71, 72] function approximation, and robotics applications [63, 73]. As described in [64, 74, 75] a multi-layer feed-forward network consists mainly of sensory units that constitute the input layer, one or more hidden layers and an output layer.

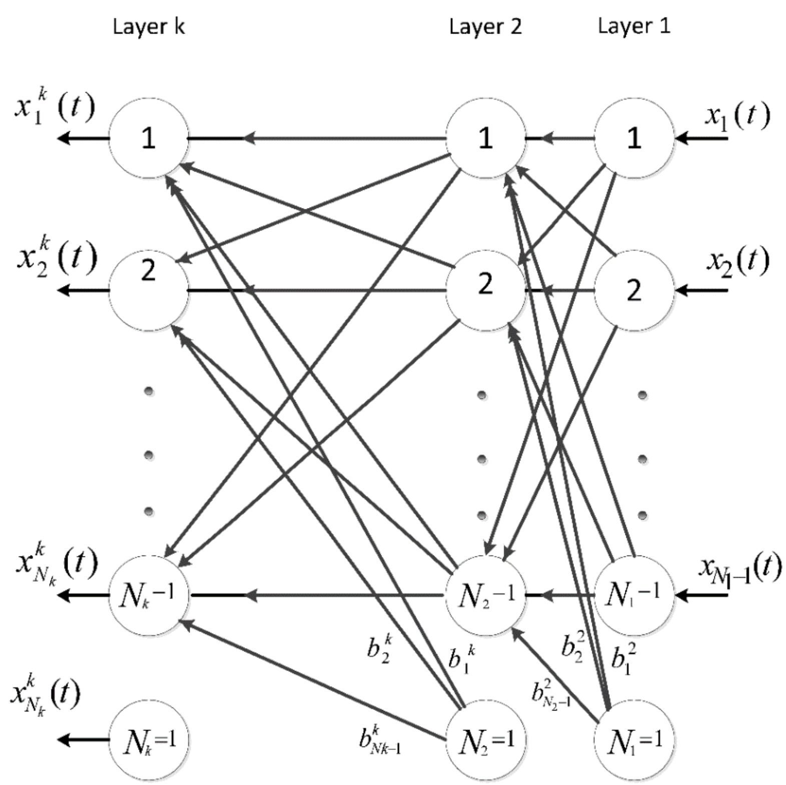


Figure 5: Schematic of feed-forward multilayer perceptron network [76]

As shown in Figure 5 and described in [74, 75], each node is connected to all nodes in the adjacent layer by links (weights), and computes a weighted sum of the inputs. An offset (bias) is added to the resultant sum followed by a nonlinear activation function application. The input signal propagates through the network in a forward direction on a layer-by-layer basis. Consequently, the network represents a static mapping between inputs and outputs.

As per [74, 75], let k denote the total number of layers, including the input and output layers. $\mathrm{Node}(n, i)$ denotes the i^{th} node in the n^{th} layer, and $N_n - 1$ is the total number of nodes in the n^{th} layer. As shown in Figure 6, the operation of $\mathrm{node}(n + 1, i)$ is described by the following equation [13]:

Equation 3.1-3.3: Feed-Foward Neural Network

$$x_i^{n+1}(t) = \varphi\left(\sum_{j=1}^{N_n-1} w_{i,j}^n\, x_j^n(t) + b_i^{n+1}\right) \tag{3.1}$$

where, $x_i^n(t)$ denotes the output of $node(n, j)$ for the t training pattern, $w_{i,j}^n$ denotes the link weight from $\mathrm{node}(n, j)$ to the $\mathrm{node}(n + 1, i)$. b_i^n is the node offset (bias) for $\mathrm{node}(n, i)$ [76].

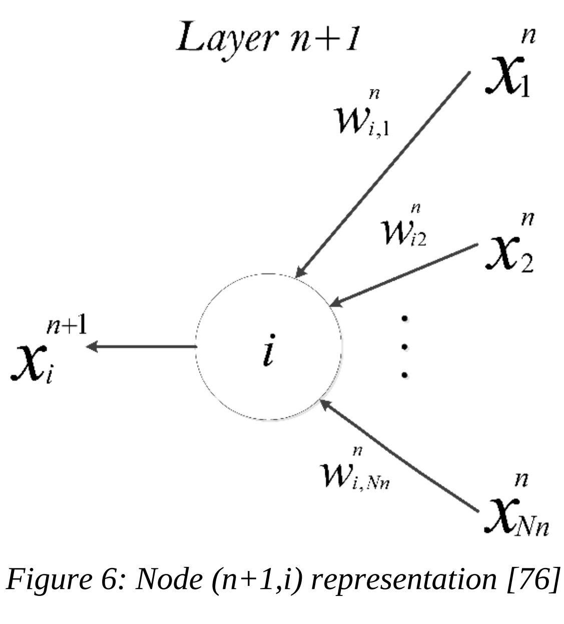


Figure 6: Node (n+1,i) representation [76]

The function $\varphi(.)$ is a nonlinear sigmoid activation function defined by [76]:

$$\varphi(w) = \frac{1}{1+e^{-aw}} \qquad a > 0 \text{ and } -\infty < w < \infty \tag{3.2}$$

For simplicity, the node bias is considered as a link weight by setting the last input N_n to $node(n+1, i)$ to the value of one as follows [76]:

$$\begin{aligned} x_{N_n}^{n}(t) &= 1, \qquad & 1 \le n \le k \\ w_{i,N_n}^{n} &= b_i^{n+1}, \qquad & 1 \le n \le k-1 \end{aligned}$$

Consequently, (3.1) can be rewritten in the following form [76]:

$$x_i^{n+1}(t) = \varphi\left(\sum_{j=1}^{N_n} w_{i,j}^{n}\, x_j^{n}(t)\right) \tag{3.3}$$

In general, raw data should not be directly applied to the input layer of a neural network. If the data given is high dimensional or contains large amounts of raw/unprocessed data, it will increase the number of weights in the neural network and slow down the training algorithm. Pre-processing the data in order to obtain important features, thus reducing the dimensionality of the training data and the number or required weights. A basic algorithm for applying neural networks to fault detection consists of using a signal processing technique to obtain a figure of merit f for the different time signals. If f is highly dimensional, a feature extraction algorithm to reduce its dimensionality with keeping most of its information content transforming f into f_e. Finally, the neural network is trained on f_e in either supervised or unsupervised mode. The following sections will discuss the figures of merits used and the accompanying pre-processing technique [63, 69].

3.3 Proposed Experimental Setup

The experimental setup consists of a scale aluminum helicopter blade (Figure 7 and 8) exposed to transverse vibratory excitation at the hub using single axis electrodynamic shaker as shown in Figure 9. The data acquisition will feature a number of hardware redundancies including accelerometers, piezo electric transducers, and strain gauges in order to improve the efficacy of the neural network training and accuracy of the outputs as shown in Figure 10. The training phase will consist of using known healthy systems in parallel with a wide range of faulty systems from damage precursors to catastrophic failure. Over the course of the study, different data processing algorithms and input features/classifiers (such as natural frequencies, modal shapes, etc.) will be used to train the neural network until satisfactory fault characterization is achieved as shown in Figure 11. Accurate fault classification is important as it will eventually be used in a hierarchical control system to apply the correct counter measures to mitigate further damage and fatigue.

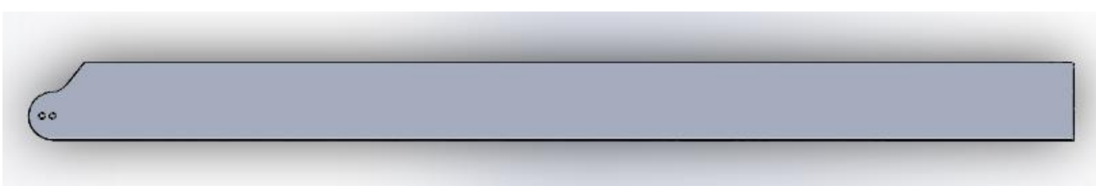


Figure 7: Top view of scale aluminum helicopter blade

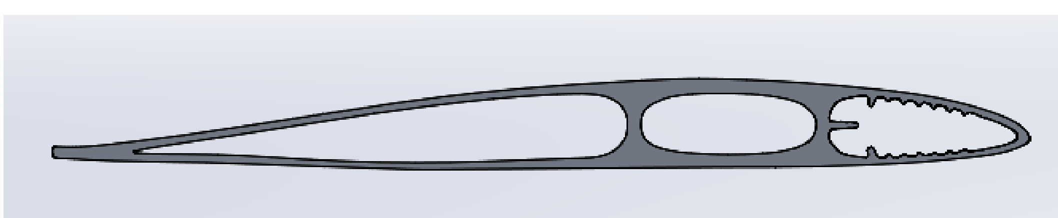


Figure 8: Cross-section of scale aluminum helicopter blade

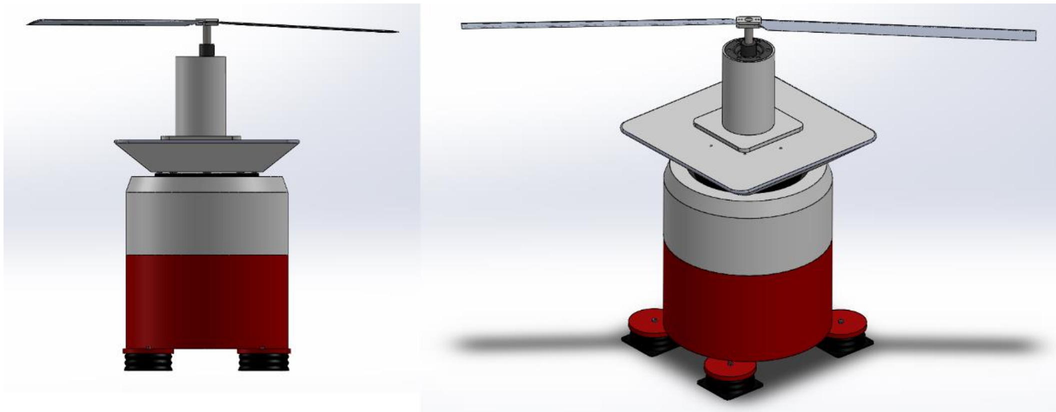


Figure 9: Scale helicopter blade and head expander fixed to the electrodynamic shaker

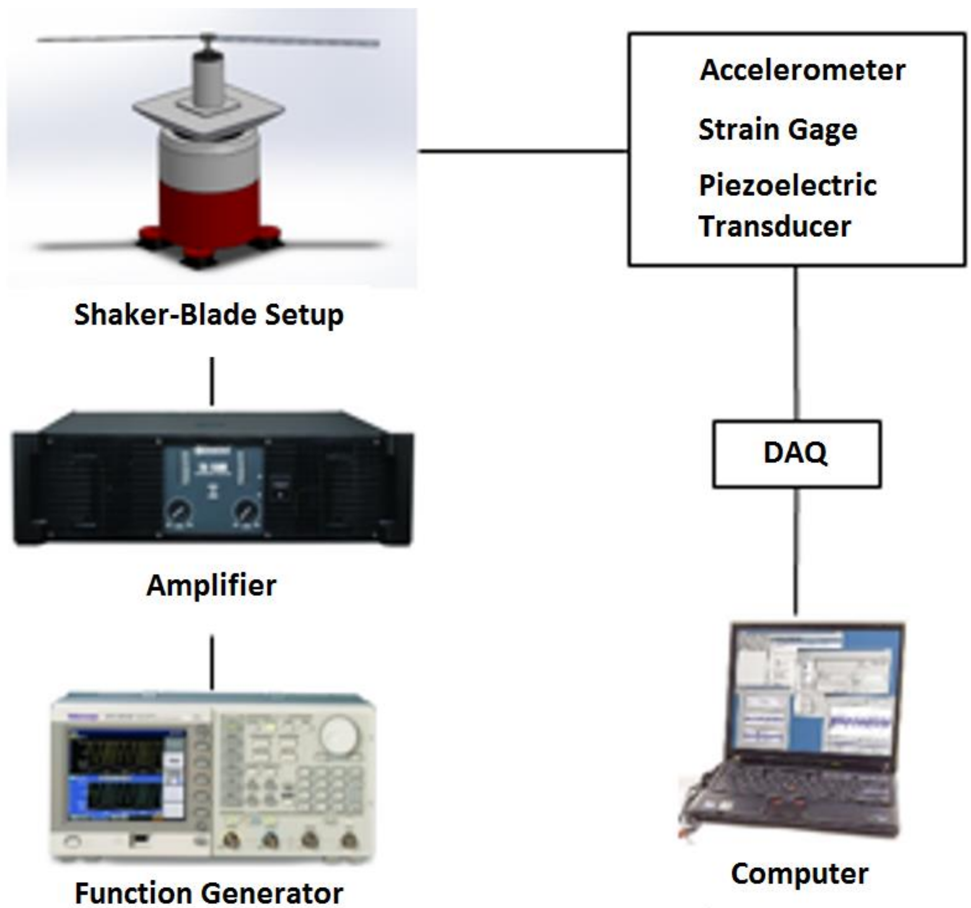


Figure 10: Proposed experimental setup

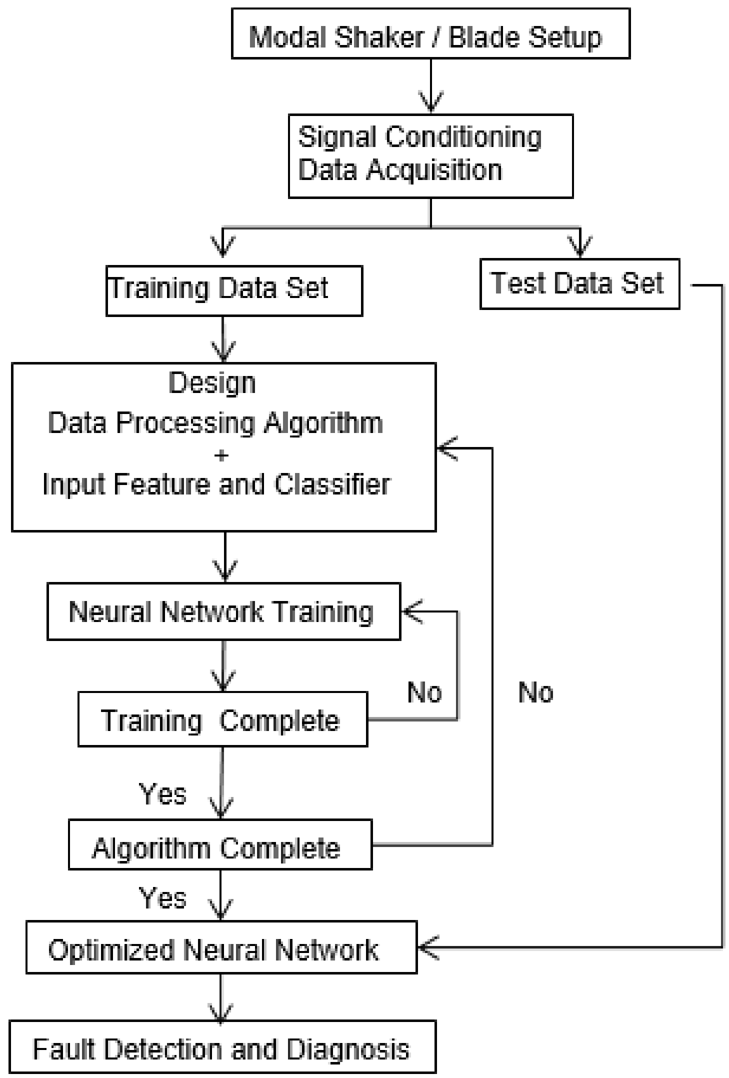


Figure 11: Neural network training flowchart

3.4 Preliminary Study

The following sections contain the preliminary study results used to justify the proposed experimental setup. The preliminary study was conducted at the University of Maryland Baltimore County while the proposed experimental setup is funded through the Army Research Lab in Aberdeen Proving Ground.

3.4.1 Methods

A preliminary study was performed on a 12” x 1” x 1/16” 6061-T6 aluminum cantilever beam as shown in Figure 12 with large relative faults in order to examine neural network strategies. The faults consisted of a ½” diameter hole located 2” from the fixed end and a ½” transverse crack located 2” from the fixed end. The beam end was deflected by ½” and released. The resulting vibration data was acquired using an ADXL335 accelerometer located at the free end of the cantilever beam and ATmega328 microcontroller as shown in Figure 13. Vibration data was collected from 100 healthy, cracked, and hole specimens in a vibration isolated environment and a non-isolated environment. The natural frequency, settling time, and damping ratio of each data set was collected. Figure 14 shows a sample of the data collected.

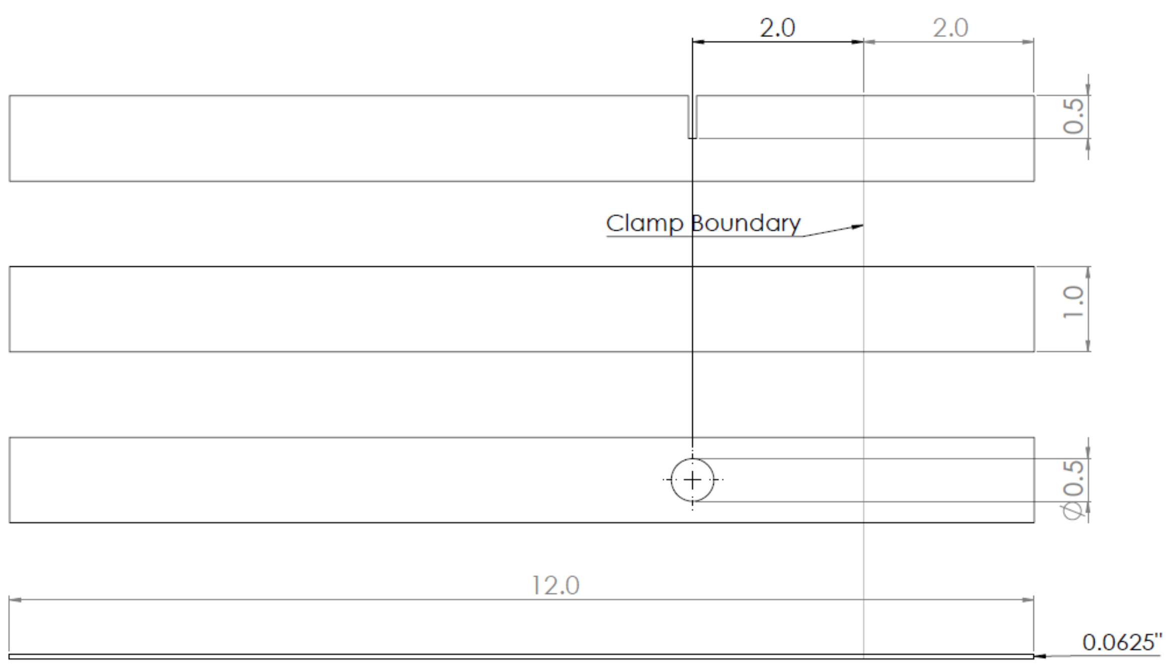


Figure 12: Engineering drawing of aluminum cantilever beam specimens. The top specimen contains a transverse crack, the middle specimen is healthy, and the bottom specimen contains a hole.

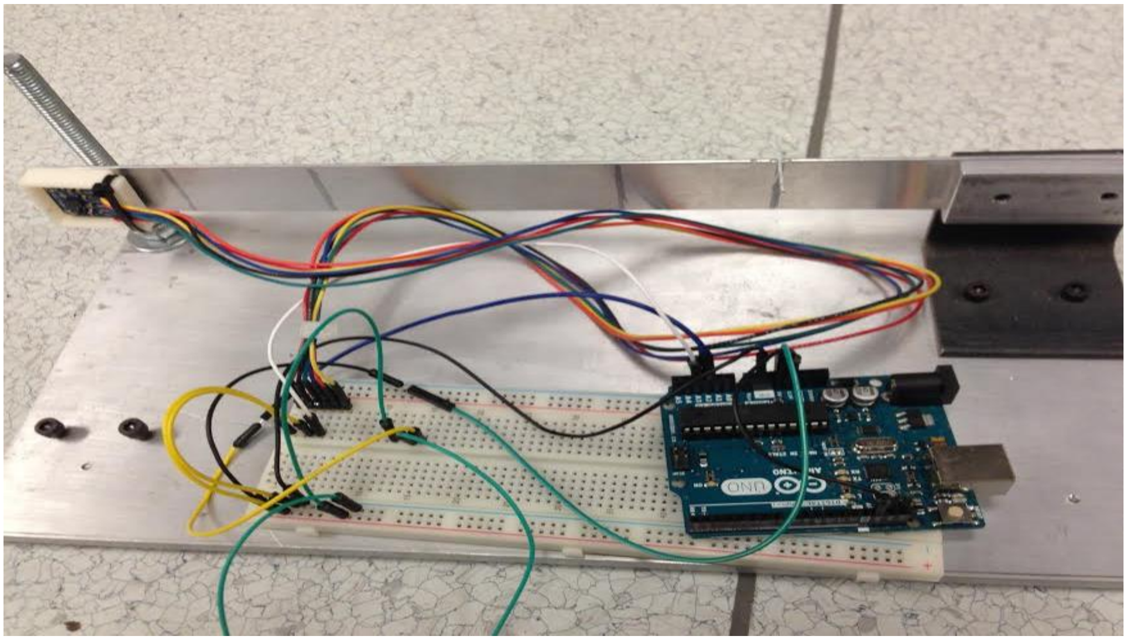


Figure 13: Experimental setup for the preliminary study

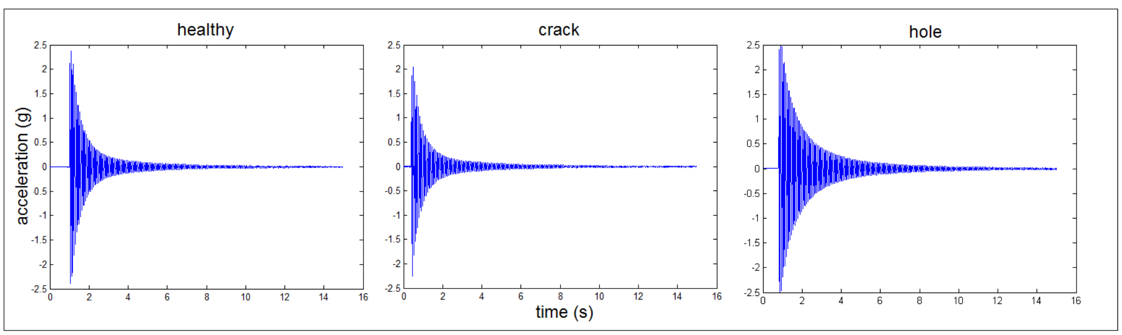


Figure 14: Vibration data collected at a sample rate of 1 kHz for healthy and faulty specimens

3.4.2 Data Pre-Processing

Raw vibration data consists of accelerometer data for a duration of approximately 15 seconds at a sampling rate of 1 kHz. Feeding this data directly into the neural network input would cause tremendous strain on the training phase. Thus, relevant characteristics such as natural frequency, settling time, and damping ratio were extracted from the data. The log decrement method was used to calculate the damping ratio where $x(t)$ is the amplitude at time t, T is the period, and n is the number of peaks away from $x(t)$.

Equation 4.1-4.2: Log decrement method for damping ratio

$$\delta = \frac{1}{n} ln \frac{x(t)}{x(t+nT)} \tag{4.1}$$

$$\xi = \frac{1}{\sqrt{1+\left(\frac{2\pi}{\delta}\right)^2}} \tag{4.2}$$

The natural frequency and second order approximation of the system were modeled using equations (5.1) and (5.2), respectively.

Equation 5.1-5.2: Second order approximation of the system

$$\omega_n = \frac{\omega_d}{\sqrt{1-\xi^2}} \tag{5.1}$$

$$\frac{Y(s)}{U(s)} = \frac{{\omega_n}^2}{s^2 + 2\xi\omega_n + \ {\omega_n}^2} \tag{5.2}$$

The normal distribution of these parameters was created in order to assess the viability of a neural network for fault detection using these properties as shown in Figure 15.

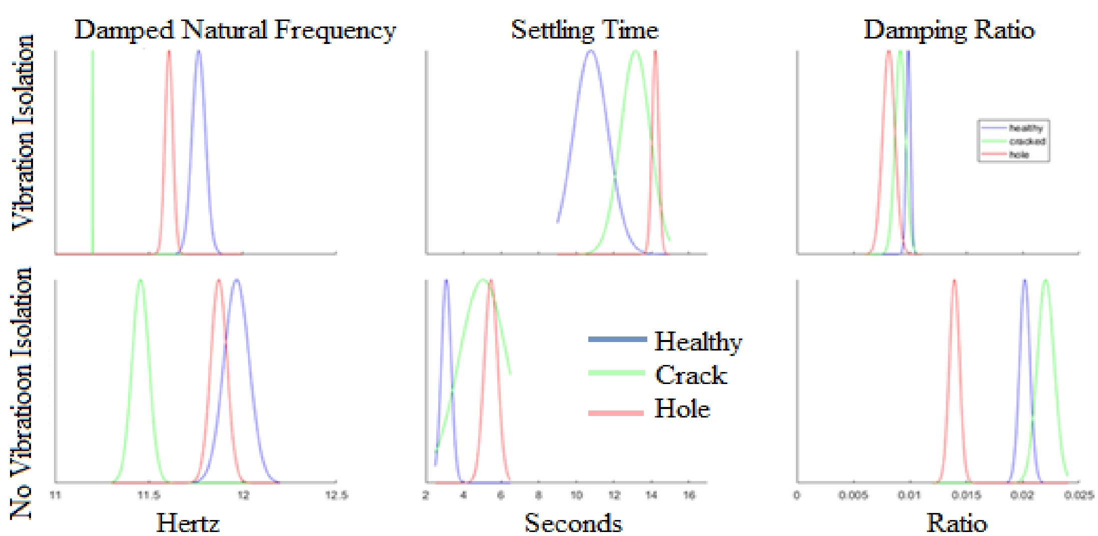


Figure 15: Normal distribution of different parameters for healthy and faulty specimens

The parameter with greatest distinction for different health states is natural frequency. While settling time and damping ratio show significant overlap in their distributions, there is still a consistent trend allowing for added redundancy.

3.4.3 Neural Network Training

The normal distributions were used to simulate 2,000 healthy samples and 2,000 samples of each type of fault (hole and crack). A supervised learning method was used to train the neural network with these samples in which the healthy and faulty samples were known. The output target classification of the known samples consisted of a 1x3 binary logic matrix in which a healthy sample was denoted by [1 0 0], a cracked sample was denoted by [0 1 0], and a sample with a hole was denoted by [0 0 1]. Figure 16 shows the basic neural network structure in with three inputs (natural frequency, settling time, and damping ratio) and an output classifying the system as being healthy, having a transverse crack, or having a hole. In order to test the neural network, an unknown sample was used and the neural network was tasked with correctly classifying the health state.

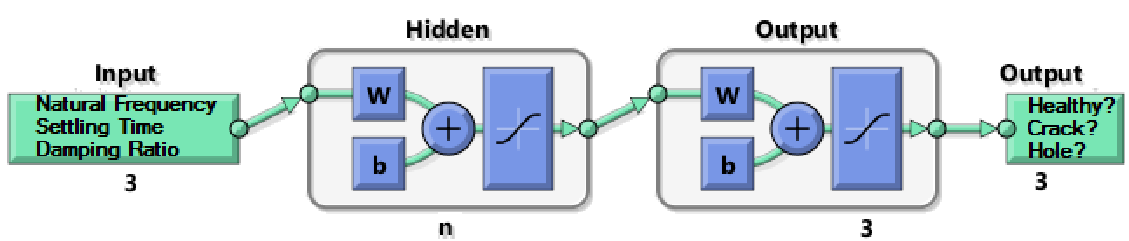


Figure 16: Basic structure of the neural network

3.4.4 Results of Preliminary Study

Hidden layers exist between the input and output layer as explained in Section 2.2.2. The connection between the hidden layers are modified through weight assignments. These weights are adjusted until classification accuracy is optimized. Increasing the number of hidden layers can increases classification accuracy because each layer adds its own level of non-linearity which cannot be contained in a single layer since each layer's inputs are linearly combined.

The number of hidden layers was varied until the neural network was able to correctly identify all test samples. The confusion matrix is a tool used to measure the performance of the ANN classification algorithm. Figure 17 shows the confusion matrices for each number of hidden layers used. In this case, the target classes 1, 2, and 3 represent the true state of the helicopter blade which are healthy, crack, and hole respectively. The output class shows the classification of each sample made by the ANN. Thus, the confusion matrix illustrates the percentage of correct output classifications for each target class. With 1 hidden layer, the ANN classified all samples as having a hole. Logically, this results in a 33.33% accuracy which is no better than a random guess.

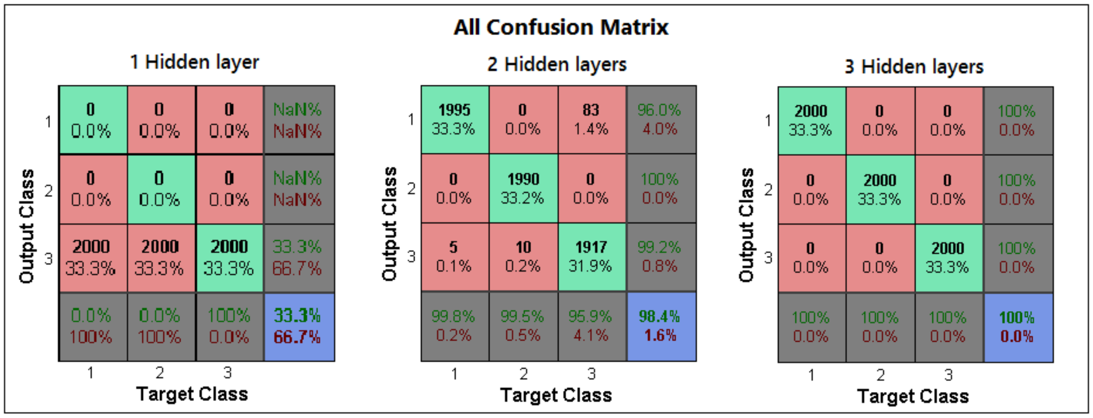


Figure 17: Confusion matrices for three types of hidden layers

Choosing the correct number of hidden layers is vital in optimizing limited computing resources. The results show that one hidden layer is insufficient for identifying the health state of the specimens. However, with two hidden layers, the neural network achieved an overall 98.4% success rate in correctly identifying the health state of the aluminum cantilever beam. At three hidden layers, the neural network was able to identify the health state with 100 percent accuracy.

An epoch refers to one forward pass and backward pass of all training samples. The weights of the ANN are adjusted after each epoch in order to minimize the classification error. After a number of epochs are performed (which can be set by the user) the ANN selects the weights associated with the local means squared error (MSE) minima. Figure 18 shows that the neural network achieved the best validation performance at epoch 60 through batch training. Even by the 30th epoch, the MSE was below 0.001.

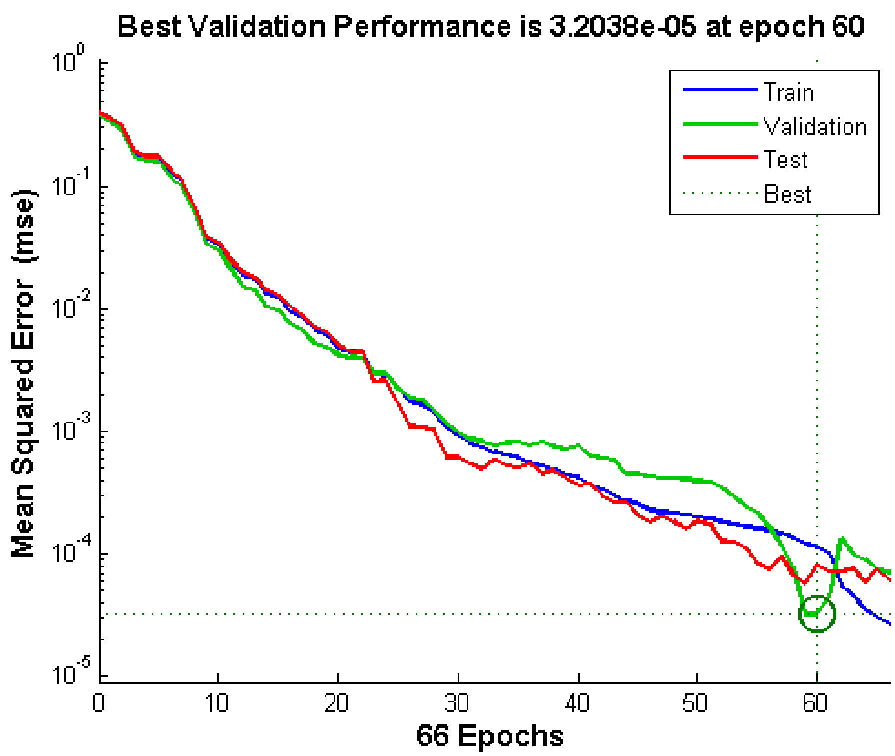


Figure 18: Validation performance at three hidden layers through minimization of the validation MSE

<u>3.5 Summary</u>

The preliminary study used natural frequency, damping ratio, and settling time extracted from vibration data. The data was collected from aluminum cantilever beams that were deemed healthy, contained a hole, and contained a transverse crack. The ANN was trained to identify the health state of the cantilever beam based on the vibration data. Despite significant overlap in the distribution of several parameters (mainly damping ratio and settling time) the ANN was able to identify the health state of the aluminum beam with high accuracy when 3 hidden layers were used. These results support the implementation of sensor redundancy to aid in ANN output accuracy.

Chapter 4: A Variable Structure-Based Strategy Applied to an Electromechanical System

This paper proposes the use of a variable structure-based estimation method, referred to the smooth variable structure filter (SVSF), in an effort to detect and identify changes or faults experienced by an electromechanical system. The proposed fault detection and diagnosis strategy is presented in Section 4.1. The electromechanical system is presented in Section 4.2, and the results are discussed in detail in Section 4.3.

4.1 Overview of Fault Detection Strategy

The partial derivative of the a *posteriori* covariance (trace) with respect to the smoothing boundary layer term ψ_{k+1} is the basis for obtaining a time-varying strategy for the specification of ψ_{k+1}. In linear systems, this smoothing boundary layer yields an optimal gain (exactly the KF) [48, 51]. Previous forms of the SVSF included a vector form of ψ, which had a single smoothing boundary layer term for each corresponding measurement error [48, 62]. Essentially, the boundary layer terms were independent of each other such that the measurement errors would not mix when calculating the corresponding gain, leading to reduced estimation accuracy. In an effort to obtain a smoothing boundary layer equation that yielded more accurate state estimates, a full smoothing boundary layer matrix was proposed in [48, 51, 77]. Hence, consider the following smoothing boundary layer form:

Equation 6.1-6.5:Smoothing Boundary Layer

$$\psi = \begin{bmatrix} \psi_{11} & \psi_{12} & \cdots & \psi_{1m} \\ \psi_{12} & \psi_{22} & \cdots & \psi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{m1} & \psi_{m2} & \cdots & \psi_{mm} \end{bmatrix} \tag{6.1}$$

This definition includes terms that relate one smoothing boundary layer to another (i.e., off-diagonal terms). To solve for the time-varying smoothing boundary layer based on (4.1), consider the following in conjunction with (1.5):

$$\frac{\partial\left(trace[P_{k+1|k+1}]\right)}{\partial\psi} = 0 \tag{6.2}$$

As described in [32], a solution for the smoothing boundary layer from (3.2) is defined as follows:

$$\psi_{k+1} = \left(\bar{E}^{-1} C_k P_{k+1|k} C_k^T S_{k+1}^{-1}\right)^{-1} \tag{6.3}$$

where S_{k+1} and A are defined respectively by:

$$S_{k+1} = C_k P_{k+1} C_k^T + R_{k+1} \tag{6.4}$$

$$E = \left(\left|e_{z_{k+1|k}}\right| + \gamma \left|e_{z_{k|k}}\right|\right) \tag{6.5}$$

This paper proposes the use of (6.3) to determine the presence of modeling uncertainty which can be detected and identified as faults. For example, as discussed previously, the width of the smoothing boundary layer provides an indicator of performance in terms of the estimation accuracy. If the width or value is small, the system used by the SVSF closely matches that of the true system. Whereas if the width is large, the system used by the SVSF does not match the true system. If a finite number of system operations or models are known, then a bank of filters can be implemented and (6.3) can be used to accurately and quickly detect and identify the correct mode of operation. This concept is illustrated in Section 4.2.

4.2 Electromechanical System and Simulation Results

In this paper, an electromechanical system based on a type of aerospace actuator was studied [48, 78]. An electrohydrostatic actuator (EHA) is typically used in the aerospace industry for aircraft maneuvering by controlling flight surfaces. EHAs are self-contained units comprised of their own pump, hydraulic circuit, and actuating cylinder [48, 62]. The main components of an EHA include a variable speed motor, an external gear pump, an accumulator, inner circuitry check valves, a cylinder (or actuator), and a bi-directional pressure relief mechanism. A mathematical model for the EHA has been described in detail in [48, 79, 51]. For the purposes of this paper, only the main state space equations will be explored. The input to the system is the rotational speed of the pump ω_p, with typical units of rad/s. In this setup, the sample rate for this simulation was defined as $T = 0.1\ ms$. The state space equations are defined as follows:

Equation 7.1-7.6: System Equations

$$x_{1,k+1} = x_{1,k} + Tx_{2,k} + Tw_{1,k} \tag{7.1}$$

$$x_{2,k+1} = x_{2,k} + Tx_{3,k} + Tw_{2,k} \tag{7.2}$$

$$x_{1,k+1} = \left[1 - T\left(\frac{BV_0 + M\beta_e L}{MV_0}\right)\right] x_{3,k} - T\frac{(A^2 + BL)\beta_e}{MV_0} x_{2,k}$$

$$- T\left[\frac{2B_2 V_0 x_{2,k} x_{3,k}}{MV_0} + \frac{\beta_e L\left(B_2 x_{2,k}^2 + B_0\right)}{MV_0}\right] sign(x_2, k) \qquad (7.3)$$

$$+ T\frac{AD_p\beta_e}{MV_0} u_k + Tw_{3,k}$$

Note that A (in this case) refers to the piston cross-sectional area, B_2 represents the load friction present in the system, β_e is the effective bulk modulus (i.e., the 'stiffness' in the hydraulic circuit), D_p refers to the pump displacement, L represents the leakage coefficient, M is the load mass (i.e., weight of the cylinders), and V_0 is the initial cylinder volume. Two more models were created based on a severe friction fault (the friction was increased 3 times) and a severe leakage fault (the leakage coefficient was increased 4 times). The normal, friction fault, and leakage fault system matrices (A_1, A_2, and A_3) are respectively defined as follows:

$$A_1 = \begin{bmatrix} 1 & 0.0001 & 0 \\ 0 & 1 & 0.0001 \\ 0 & -41.0258 & 0.6099 \end{bmatrix} \qquad (7.4)$$

$$A_2 = \begin{bmatrix} 1 & 0.0001 & 0 \\ 0 & 1 & 0.0001 \\ 0 & -51.8627 & 0.2226 \end{bmatrix} \qquad (7.5)$$

$$A_3 = \begin{bmatrix} 1 & 0.0001 & 0 \\ 0 & 1 & 0.0001 \\ 0 & -73.5364 & 0.6015 \end{bmatrix} \qquad (7.6)$$

Note that all three input gain matrices remained the same, and were calculated as follows:

$$B = \begin{bmatrix} 0 \\ 0 \\ 0.0135 \end{bmatrix} \tag{7.7}$$

Note also that artificial system and measurement noise was added to the simulation problem to make it more challenging. The zero-mean Gaussian noise was generated using system and measurement noise covariance's Q and R which were diagonal matrices with elements equal to 1×10^{-6}. The desired position, velocity, and acceleration trajectories are shown in the following three figures (Figure 19-21).

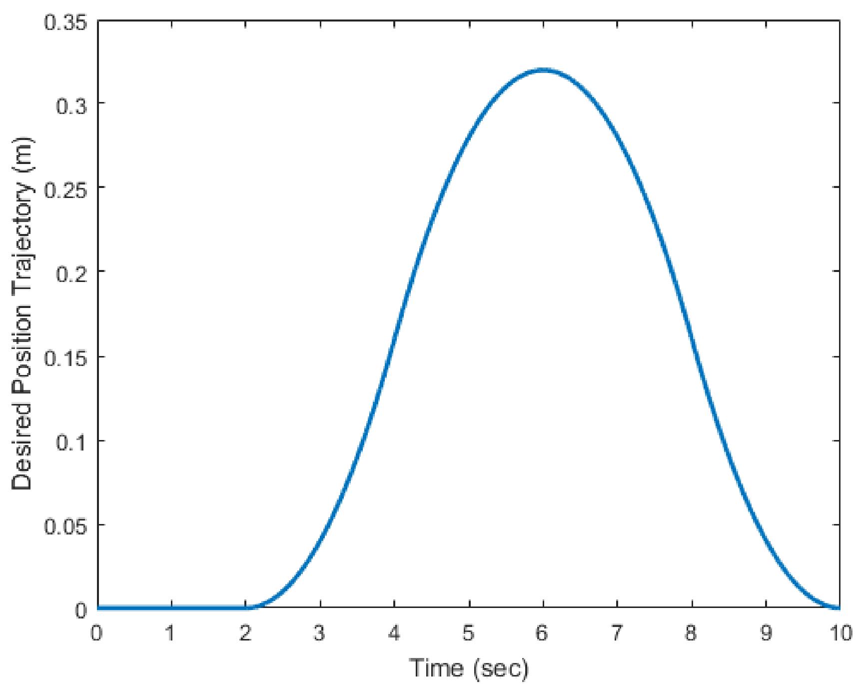


Figure 19: Desired EHA position trajectory

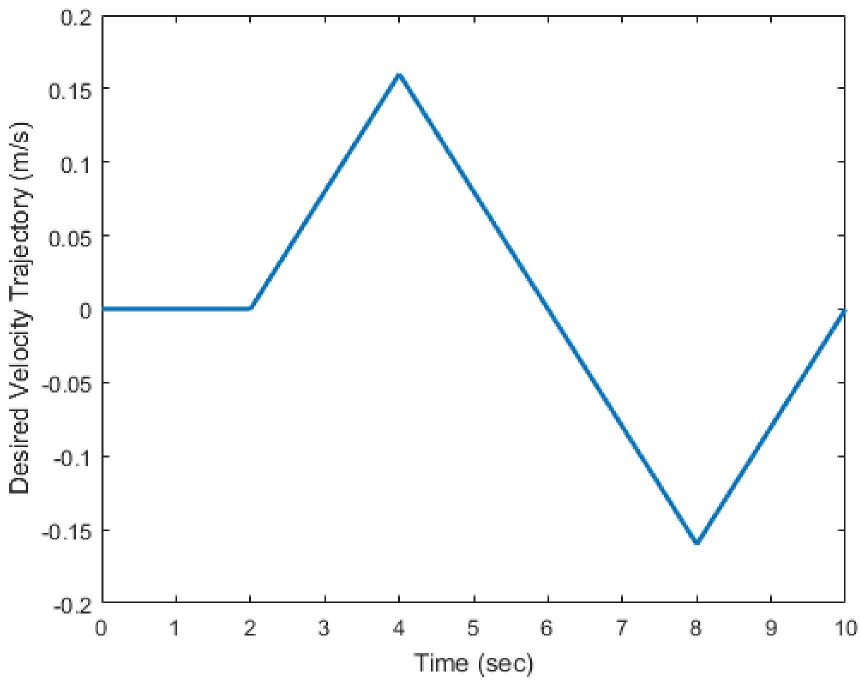


Figure 20: Desired EHA velocity trajectory

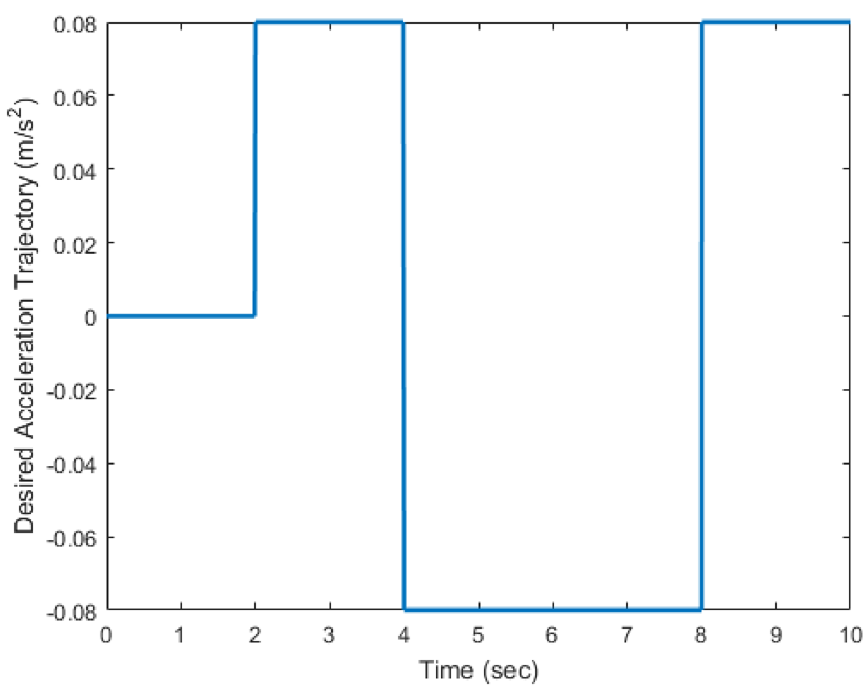


Figure 21: Desired EHA acceleration trajectory

Note that for the first 4 seconds, the system behaved normally. A friction fault was injected at 4 seconds and lasted for 4 seconds. At 8 seconds, the friction fault was remove and a 2 second leakage fault was implemented.

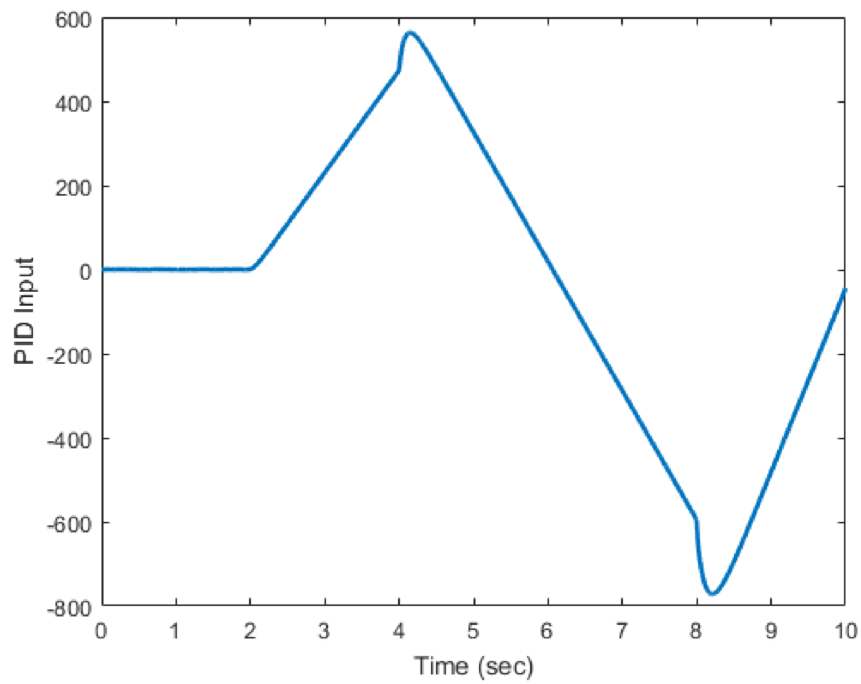


Figure 22: Calculated system input from PID controller

4.3 Estimation and Fault Detection Results

The following three figures (Figs. 23-25) show the results of applying the KF and SVSF estimation strategies on the aforementioned electromechanical system. The estimation results for the KF and SVSF were nearly identical for the first two states. However, for the third state, the KF estimate was slightly noisier, whereas the SVSF smoothed out the estimated acceleration.

Figure 26 illustrates the acceleration state boundary layer values for each mode of operation. Recall that the system behaved normally for the first four seconds (A_1), followed by a four second friction fault (A_2), and finally a two second leakage fault (A_3). The magnitude of the calculated boundary layer, based on the SVSF gain and state error covariance matrix, provides a method for detecting system changes. A small magnitude indicates that the system behaviour closely matches the model used by the SVSF. Therefore, the Psi_{A_1} term is expected to be smaller than the other two boundary layer terms for the first four seconds, and is verified in Figure 26. During the next four sections, the system operates in the presence of a friction fault (A_2), and the Psi_{A_2} term is found to be the smallest of the three. Finally, the system operates with leakage (A_3), and is verified since the magnitude of the Psi_{A_3} term is the smallest. The time-varying boundary layer that was derived in Section 4.1 is shown to be a viable term for fault detection and diagnosis. However, this method requires that the system behaves according to a finite number of models that the user or engineer can describe mathematically.

Another interesting property of the SVSF is observed when a spectrogram of the acceleration boundary layer values is created for each of the three modes of operations (Figs. 27-29). A spectrogram is a visual representation of the spectrum of frequencies in a signal as they vary with time or some other variable. In this case, the signal is based on the calculated time-varying boundary layer. Visual patterns appear in each figure based on the ratio of power and frequency (dB/Hz). Similar to the study of the boundary layer magnitudes, in this case, the smaller the ratio the better match in terms of system operation and the model used by the SVSF.

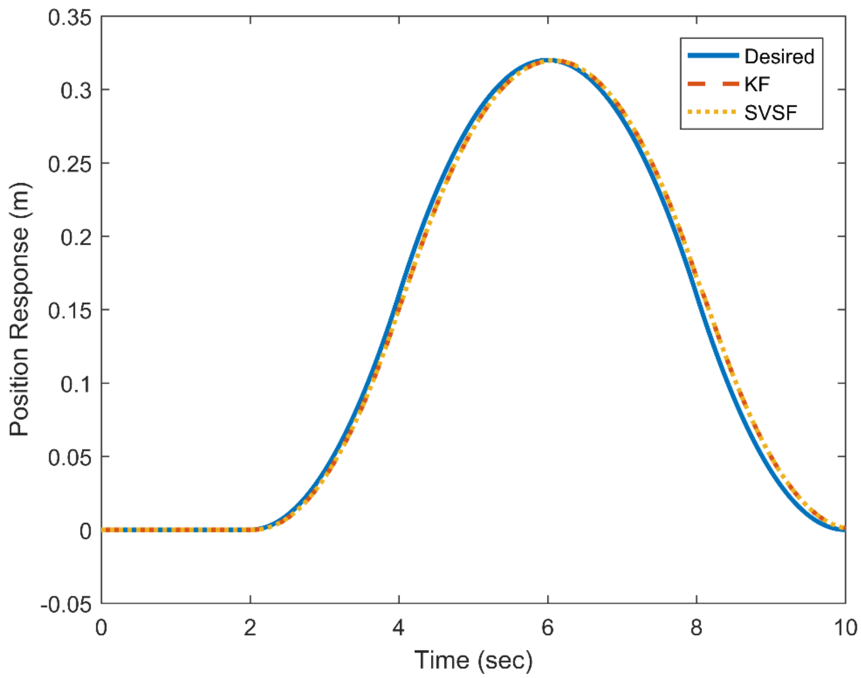


Figure 23: Estimated position trajectory using the KF and SVSF strategies

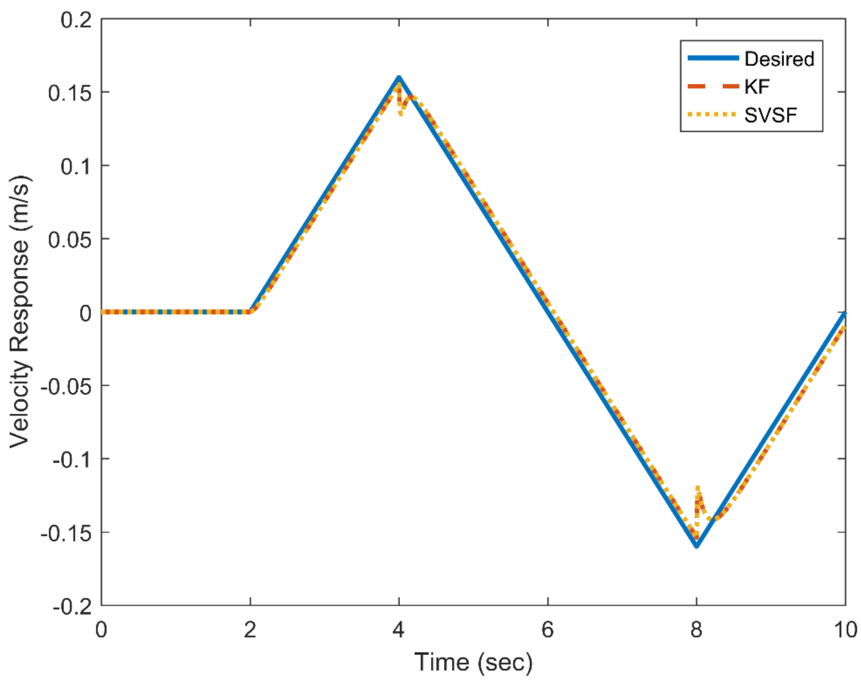


Figure 24: Estimated velocity trajectory using the KF and SVSF strategies

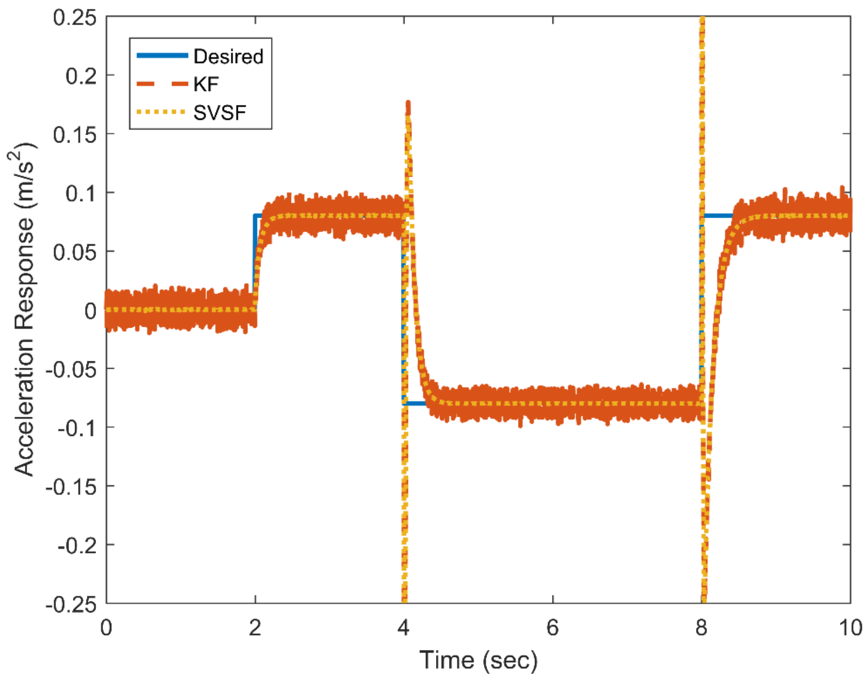


Figure 25: Estimated acceleration trajectory using the KF and SVSF strategies

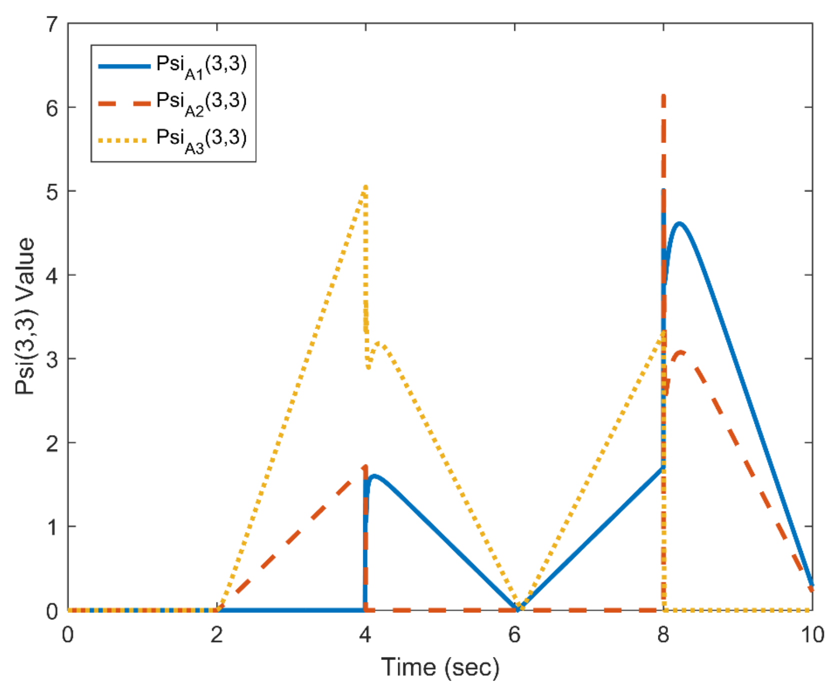


Figure 26: Acceleration state boundary layer values for each mode of operation

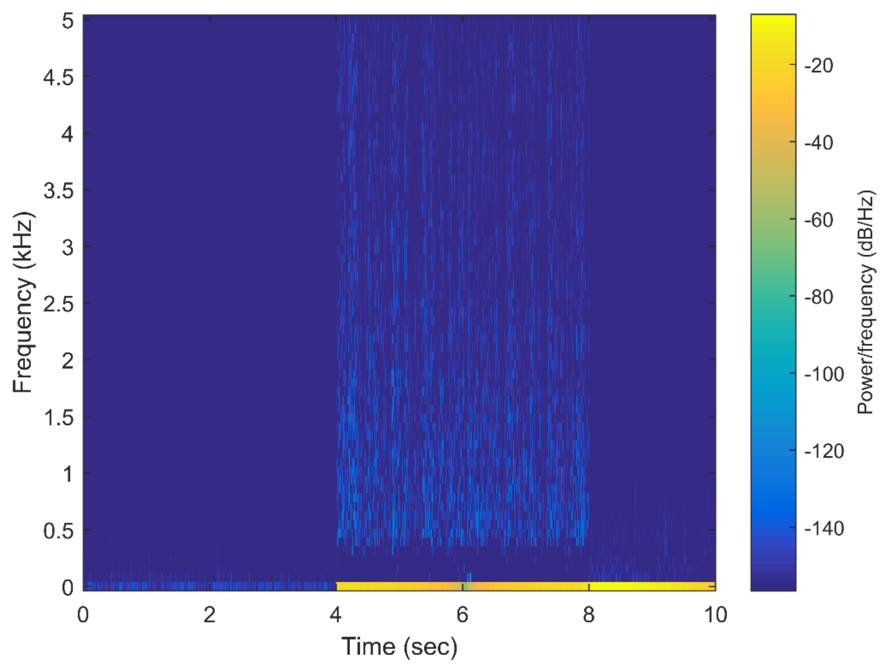


Figure 27: Spectrogram of the acceleration boundary layer values using the normal system model

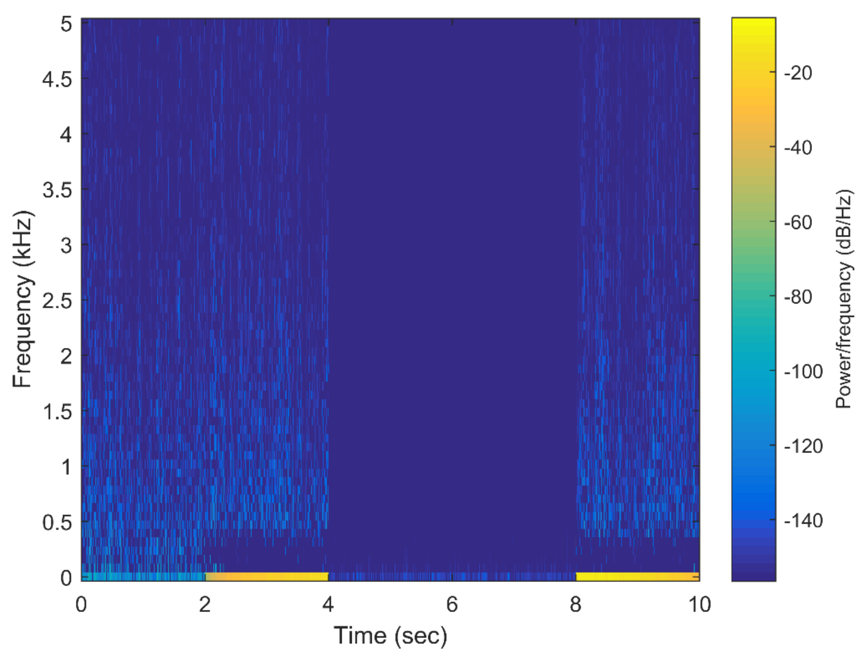


Figure 28: Spectrogram of the acceleration boundary layer values using the friction fault model

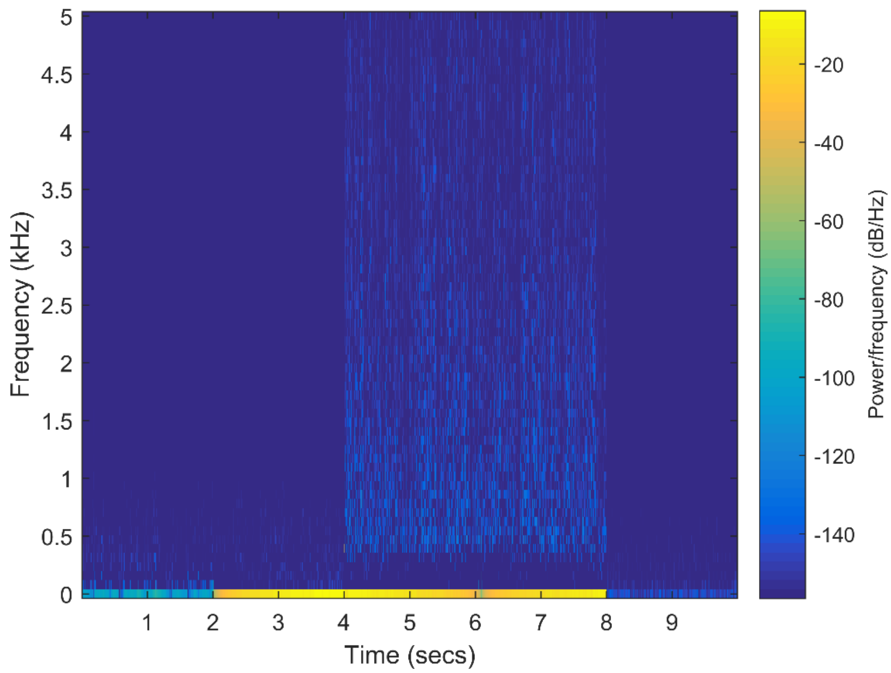


Figure 29: Spectrogram of the acceleration boundary layer values using the leakage fault model

Figure 30 is a combination of Figs. 27-29 at low frequencies (less than $35\ Hz$). This figure clearly shows the presence of faults (low vs high power and frequency ratio). Based on knowledge of the system, the correct operating mode can easily be identified. For example, as per Psi_{A_1}, the system is shown to operate normally for four seconds, and then abnormally for the remainder of the simulation. The leakage fault (A_3) is shown to exist between 8 and 10 seconds, as per Psi_{A_3}.

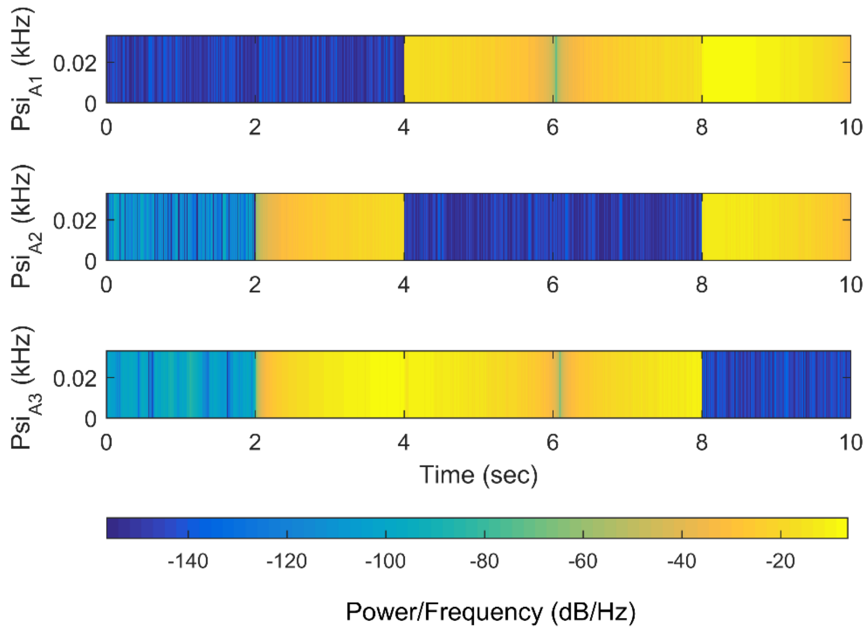


Figure 30: Low frequency spectrogram of the acceleration boundary layers for all three models

4.4 Summary

The smooth variable structure filter (SVSF) is a sub-optimal filter but is considerably more robust than the KF [48]. In this paper, properties of the SVSF were explored in an effort to detect and diagnosis faults in an electromechanical system [48]. It was determined that the definition for the time-varying smoothing boundary layer may be used to accurately and quickly detect and identify changes in a system [48].

Chapter 5: Concluding Remarks

5.1 Summary of Thesis

The ultimate goal of FDI is to maximize the life-span of equipment and minimize the cost of maintenance [1]. The development of intelligent diagnostic, prognostic, and health management technology has proven to be important for industrial and defense maintenance procedures in recent years, particularly in aerospace. While diagnostic technology for aircraft have existed for more than 50 years, modern CPUs permit on-board intelligent and estimation-based FDI methods. This thesis discussed two strategies in particular: artificial neural networks and smooth variable structure filters (SVSF). The purpose of this thesis is to propose a method of health state awareness for a helicopter blade using an artificial neural network as well as develop a variable structure-based fault detection and diagnosis strategy for an electromechanical actuator [4, 5].

The proposed experimental setup for a helicopter blade uses a single axis electrodynamic shaker to subject a scale aluminum helicopter blade to transverse vibratory excitation at the hub [4]. Data collected from an array of accelerometers, piezo electric transducers, and strain gauges embedded along the blade would be used to train a neural network for fault detection and diagnosis. The training phase would include healthy samples as well as samples with known fault types and locations. In order to test the efficacy of the proposed setup, a preliminary study was performed using an accelerometer adhered to an aluminum cantilever beam. Vibrational data was recorded after the cantilever beam was released from an initial deflection.

The preliminary study used natural frequency, damping ratio, and settling time extracted from vibration data. The data was collected from aluminum cantilever beams that were deemed healthy, contained a hole, and contained a transverse crack. The ANN was trained to identify the health state of the cantilever beam based on the vibration data. Despite significant overlap in the distribution of several parameters (mainly damping ratio and settling time) the ANN was able to identify the health state of the aluminum beam with high accuracy when 3 hidden layers were used. These results support the implementation of sensor redundancy to aid in ANN output accuracy.

This thesis also discusses properties of the SVSF in an effort to detect and diagnosis faults in an electromechanical system. In this case, an electromechanical system was based on a type of aerospace actuator [33]. An electrohydrostatic actuator (EHA) is typically used in the aerospace industry for aircraft maneuvering by controlling flight surfaces. The SVSF was compared to the KF and showed nearly identical results for the trajectory and velocity states. However, for the acceleration state the KF estimate was slightly noisier while the SVSF smoothed out the estimated acceleration. It was determined that the definition for the time-varying smoothing boundary layer may be used to accurately and quickly detect and identify changes in a system. Thus, SVSF displays a competitive advantage over KF for this application.

5.2 Future Work

In the case of the state health awareness using an ANN, this study only used highly exaggerated faults for classification. In additional, only three parameters were used to train an ANN. Finally, the study incorporated only one sensor (accelerometer) in a single location. Future work will include training the neural network with these additional parameters as well as a wide range of fault types, locations, and sizes in order develop a hierarchical control system that assesses the prognosis of the fault and applies proper counter measures in order to prevent further fault propagation and exacerbation. In order to achieve this, a highly redundant hardware system consisting of several sensors (strain gauges, piezo electric transducers, accelerometers, and fiber optics) will be used to characterize faults but also predict propagation.

Although the SVSF is a sub-optimal filter it is considered far more robust than KF. This thesis examined the application of a SVSF to dynamic model of an EHA. Future work includes the application of the proposed methodology on an experimental setup and a comparison of the result with other FDI techniques.

Appendices

Appendix A: Experimental Equipment

A.1 ADXL335 Accelerometer

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

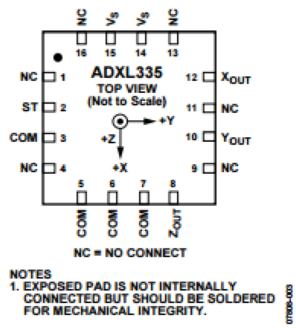


Figure 2. Pin Configuration

Table 3. Pin Function Descriptions

Pin No.	Mnemonic	Description
1	NC	No Connect[1].
2	ST	Self-Test.
3	COM	Common.
4	NC	No Connect[1].
5	COM	Common.
6	COM	Common.
7	COM	Common.
8	Z_{OUT}	Z Channel Output.
9	NC	No Connect[1].
10	Y_{OUT}	Y Channel Output.
11	NC	No Connect[1].
12	X_{OUT}	X Channel Output.
13	NC	No Connect[1].
14	V_S	Supply Voltage (1.8 V to 3.6 V).
15	V_S	Supply Voltage (1.8 V to 3.6 V).
16	NC	No Connect[1].
EP	Exposed Pad	Not internally connected. Solder for mechanical integrity.

[1]NC pins are not internally connected and can be tied to COM pins, unless otherwise noted.

SPECIFICATIONS

T_A = 25°C, V_S = 3 V, C_X = C_Y = C_Z = 0.1 μF, acceleration = 0 *g*, unless otherwise noted. All minimum and maximum specifications are guaranteed. Typical specifications are not guaranteed.

Table 1.

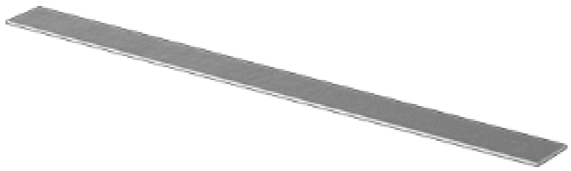
Parameter	Conditions	Min	Typ	Max	Unit
SENSOR INPUT	Each axis				
Measurement Range		±3	±3.6		*g*
Nonlinearity	% of full scale		±0.3		%
Package Alignment Error			±1		Degrees
Interaxis Alignment Error			±0.1		Degrees
Cross-Axis Sensitivity[1]			±1		%
SENSITIVITY (RATIOMETRIC)[2]	Each axis				
Sensitivity at X_{OUT}, Y_{OUT}, Z_{OUT}	V_S = 3 V	270	300	330	mV/*g*
Sensitivity Change Due to Temperature[3]	V_S = 3 V		±0.01		%/°C
ZERO *g* BIAS LEVEL (RATIOMETRIC)					
0 *g* Voltage at X_{OUT}, Y_{OUT}	V_S = 3 V	1.35	1.5	1.65	V
0 *g* Voltage at Z_{OUT}	V_S = 3 V	1.2	1.5	1.8	V
0 *g* Offset vs. Temperature			±1		mg/°C
NOISE PERFORMANCE					
Noise Density X_{OUT}, Y_{OUT}			150		μ*g*/√Hz rms
Noise Density Z_{OUT}			300		μ*g*/√Hz rms
FREQUENCY RESPONSE[4]					
Bandwidth X_{OUT}, Y_{OUT}[5]	No external filter		1600		Hz
Bandwidth Z_{OUT}[5]	No external filter		550		Hz
R_{FILT} Tolerance			32 ± 15%		kΩ
Sensor Resonant Frequency			5.5		kHz
SELF-TEST[6]					
Logic Input Low			+0.6		V
Logic Input High			+2.4		V
ST Actuation Current			+60		μA
Output Change at X_{OUT}	Self-Test 0 to Self-Test 1	−150	−325	−600	mV
Output Change at Y_{OUT}	Self-Test 0 to Self-Test 1	+150	+325	+600	mV
Output Change at Z_{OUT}	Self-Test 0 to Self-Test 1	+150	+550	+1000	mV
OUTPUT AMPLIFIER					
Output Swing Low	No load		0.1		V
Output Swing High	No load		2.8		V
POWER SUPPLY					
Operating Voltage Range		1.8		3.6	V
Supply Current	V_S = 3 V		350		μA
Turn-On Time[7]	No external filter		1		ms
TEMPERATURE					
Operating Temperature Range		−40		+85	°C

A.2 McMaster Aluminum

McMASTER-CARR.

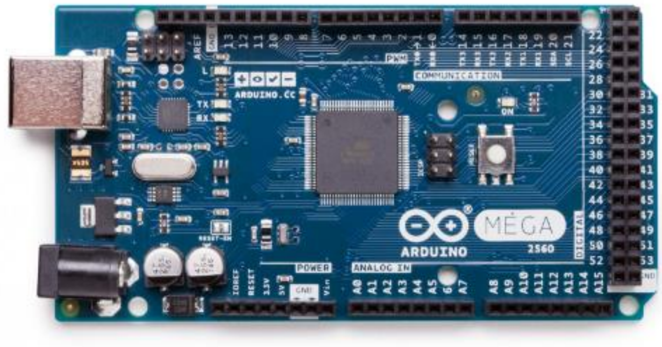
6061 Aluminum with Certification

1/16" Thick x 1" Wide



Material	6061 Aluminum
Cross Section Shape	Rectangle
Construction	Solid
Thickness	1/16"
Thickness Tolerance	-0.006" to 0.006"
Tolerance Rating	Standard
Width	1"
Width Tolerance	-0.014" to 0.014"
Yield Strength	35,000 psi
Fabrication	Heat Treated
Temper	T6
Temper Rating	Hardened
Hardness	Brinell 95
Hardness Rating	Soft
Certification	Material Certificate with Traceable Lot Number and Test Report
Heat Treatable	Yes
Appearance	Plain
Temperature Range	-320° to 300° F
Specifications Met	ASTM B221
Straightness Tolerance	Not Rated
Magnetic Properties	Nonmagnetic
Density	0.1 lbs./cu. in.
Surface Resistivity	25 Ohm-Cir Mil/ft
Melting Point Temperature	1080° F
Modulus of Elasticity	10.0 ksi × 10^3
Thermal Conductivity	1,160 Btu/hr. × in./sq. ft./°F @ 75° F
Elongation	12.5%
Material Composition	
Aluminum	95.1-98.2%
Chromium	0.4-0.8%
Copper	0.05-0.4%
Iron	0-0.7%
Magnesium	0.8-1.2%
Manganese	0-0.15%
Nickel	0-0.05%
Silicon	0.4-0.8%
Titanium	0-0.15%
Zinc	0-0.25%
Zirconium	0-0.25%
Other	0.15%
Length Tolerance	-1" to 1"
RoHS	Compliant

A.3 Arduino Mega



TECH SPECS

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Appendix B: Code and Software

B.1 Arduino

```
//connect 3.3v to AREF

const int ap1 = A5;
const int ap2 = A4;
const int ap3 = A3;

//const int ap4 = A2; ------

long t;
int sv1 = 0;
int ov1 = 0;
int sv2 = 0;
int ov2= 0;
int sv3 = 0;
int ov3= 0;
//int sv4=0; ----
//int ov4=0; -----

void setup() {
  // initialize serial communications at 9600 bps:
  Serial.begin(115200);
}

void loop() {
  analogReference(EXTERNAL);    //connect 3.3v to AREF
  // read the analog in value:
  //sv1 = analogRead(ap1);
  // map it to the range of the analog out:
  //ov1 = map(sv1, 0, 1023, 0, 255);
  // change the analog out value:
  //delay(2);
  //
  //sv2 = analogRead(ap2);

  //ov2 = map(sv2, 0, 1023, 0, 255);
 //
  //delay(2);
  //
  sv3 = analogRead(ap3);
  //delay(1);
  //ov3 = map(sv3, 0, 1023, 0, 255);

  //sv4 = analogRead(ap4); ------
```

```
 //ov4 = map(sv4, 0, 1023, 0, 255);
 t=micros();

 // print the results to the serial monitor:
 //Serial.print("Xsensor1 = " );
 //Serial.print(sv1);
 //Serial.print(" ");
 //Serial.print("\t output1 = ");
 //Serial.println(ov1);
 //Serial.print("Ysensor2 = " );
 //Serial.print(sv2);
 //Serial.print("\t output2 = ");
 //Serial.println(ov2);
 //Serial.print("Zsensor3 = " );
 Serial.print(t);
 Serial.print(" ");
 //Serial.print(sv4);
 //Serial.print(" ");
 Serial.println(sv3);
 //Serial.print("\t output3 = ");
 //Serial.println(ov3);

 delay(1);
if (t>7500000) Serial.end();
}
```

B.2 Matlab

```
%%%%% Data for Training Neural Network
%%%%% Floor Setup

clc
clear

n=10;
hidden=100;

healthy1=[11.7996   11.1699    0.0098
   11.7330    9.6547    0.0100
   11.7997   11.3232    0.0096
   11.8001   10.6714    0.0098
   11.7330   12.9765    0.0095
   11.7332    9.9133    0.0099
   11.7998   10.5009    0.0099
   11.7999   10.4529    0.0099
   11.7332   10.8375    0.0102
   11.7328   10.3644    0.0099];

crack1=[11.1998   12.5001    0.0099
   11.1997   12.5027    0.0094
   11.2001   13.8287    0.0097
   11.1998   14.1876    0.0096
   11.1998   13.2974    0.0088
   11.1999   13.3770    0.0087
   11.1996   13.0295    0.0091
   11.1996   14.2623    0.0086
   11.1996   11.8276    0.0090
   11.1996   12.8100    0.0089];

hole1=[ 11.6137   13.8732    0.0077
   11.6659   14.2131    0.0067
   11.5994   14.0349    0.0086
   11.5997   14.1925    0.0084
   11.5997   14.2886    0.0084
   11.5996   14.2815    0.0083
   11.5999   14.4564    0.0083
   11.5997   14.3634    0.0081
   11.5997   14.3754    0.0084
   11.5997   14.1121    0.0083];


%Table Setup

healthy2=[11.8999    2.7220    0.0187
   11.9976    3.0341    0.0174
   11.9974    3.0993    0.0174
   11.9976    2.9965    0.0190
   11.9974    2.9903    0.0175
   11.7977    3.3274    0.0192
   11.9976    3.1578    0.0189
```

```
   11.9974    2.6809    0.0189
   11.9972    3.5067    0.0190
   11.9976    3.3338    0.0187];

crack2=[  11.4989    3.3197    0.0170
   11.4667    6.2697    0.0175
   11.4668    3.9969    0.0173
   11.4668    3.2875    0.0189
   11.4667    3.4212    0.0171
   11.4669    6.8833    0.0153
   11.3313    5.9972    0.0172
   11.4665    5.7939    0.0171
   11.4646    5.3466    0.0190
   11.4668    5.4603    0.0171];

hole2=[ 11.9002    5.1050    0.0137
   11.7991    5.2866    0.0135
   11.8990    5.3700    0.0138
   11.8990    5.7878    0.0138
   11.7990    5.6183    0.0143
   11.8399    6.2163    0.0143
   11.8991    5.0269    0.0142
   11.8984    5.2124    0.0139
   11.8991    5.1200    0.0139
   11.8990    5.4607    0.0142];

healthy1_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    healthy1_perm(100*(i-1)+10*(j-1)+k,:)=[healthy1(i,1)
healthy1(j,2) healthy1(k,3)];
end
end
end

crack1_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    crack1_perm(100*(i-1)+10*(j-1)+k,:)=[crack1(i,1) crack1(j,2)
crack1(k,3)];
end
end
end

hole1_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    hole1_perm(100*(i-1)+10*(j-1)+k,:)=[hole1(i,1) hole1(j,2)
hole1(k,3)];
end
end
end
```

```
healthy2_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    healthy2_perm(100*(i-1)+10*(j-1)+k,:)=[healthy2(i,1)
healthy2(j,2) healthy2(k,3)];
end
end
end

crack2_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    crack2_perm(100*(i-1)+10*(j-1)+k,:)=[crack2(i,1) crack2(j,2)
crack2(k,3)];
end
end
end

hole2_perm=zeros(1000:3);
for i=1:10
for j=1:10
for k=1:10
    hole2_perm(100*(i-1)+10*(j-1)+k,:)=[hole2(i,1) hole2(j,2)
hole2(k,3)];
end
end
end




healthy1_nn = abs(healthy1_perm(randperm(n),:)-
healthy1_perm(randperm(n),:));
healthy2_nn = abs(healthy2_perm(randperm(n),:)-
healthy2_perm(randperm(n),:));
crack1_nn = abs(healthy1_perm(randperm(n),:)-
crack1_perm(randperm(n),:));
crack2_nn = abs(healthy2_perm(randperm(n),:)-
crack2_perm(randperm(n),:));
hole1_nn = abs(healthy1_perm(randperm(n),:)-
hole1_perm(randperm(n),:));
hole2_nn = abs(healthy2_perm(randperm(n),:)-
hole2_perm(randperm(n),:));

target1=[ones(2*n,1),zeros(2*n,1),zeros(2*n,1)]; %healhty
target2=[zeros(2*n,1),ones(2*n,1),zeros(2*n,1)];  %crack
target3=[zeros(2*n,1),zeros(2*n,1),ones(2*n,1)];%hole

input_data=[healthy1_nn;healthy2_nn;crack1_nn;crack2_nn;hole1_nn;hol
e2_nn];
target=[target1;target2;target3];
```

```
%%
% Neural Network

x=input_data';
t=target';
net = patternnet(hidden); %number of hidden layers
[net,tr] = train(net,x,t);
nntraintool
%view(net)





%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%Parameter Extraction%


load('new_healhty2')
flagForPlot=1;
time = a(:,1)*1e-6;
accel = a(:,2);
accel = accel - mean(accel);%offsets data to be zeroed;
accel = accel/512*3;%converst from voltage to acceleration
%accel*9.81 to get m/s^2
if flagForPlot == 1
    figure
    plot(time,accel)
    xlabel('time (s)','FontSize',18)
    ylabel('acceleration (g)','FontSize',18)
end

T=time(end)-time(1); %record time length
n=length(time); %No. of Sampling points
%note if n is odd warning will occur
t=linspace(0,T,n); %time series
Y=fft(accel,n); %Complex FFT data
Py=abs(Y); %Magnitude of the FFT
f=1/T*[0:1:n/2-1]; %Frequency in Hz

if flagForPlot == 1
    figure
    plot(f,Py(1:n/2))% plots the data
    title('FFT damped Natural Frequency','FontSize',18);
    xlabel('Frequency in Hertz','FontSize',18);
end

[m I] = max(Py(1:n/2));

dampedNatFreq = f(I);
%paramMat(i,1) = dampedNatFreq;
```

```
index_step_time = find(accel > 0.1 | accel < -0.1);%finds time where
step is applied it
index_settling_time = find(accel > 0.02 | accel < -0.02);%time
settling occurs at
settling_time = time(index_settling_time(end)) -
time(index_step_time(1));
%paramMat(i,2) = settling_time;

[m1 I1] = max(accel);%finds first max

npeak=100;
index_peaks = find(time >= time(I1) + npeak*1/dampedNatFreq -
1/dampedNatFreq*.1);
%finds subsequent peaks, 60 peaks after first largest
[m2 I2] = max(accel(index_peaks));%finds peak after the largest

delta = 1/npeak*log(m1/m2);%uses largest max and next largest max
dampRatio=delta/(4*pi^2+delta^2)^.5;%calculates the dampening ratio
%paramMat(i,3)=dampRatio;
%paramMat(i,4)=dampedNatFreq/sqrt(1-dampRatio^2);
%disp(i)
%tf(paramMat(i,4)^2,[1,2*dampRatio*paramMat(i,4),paramMat(i,4)^2])

fprintf('Damped Nat Frequency: %f, Settling Time: %f, Damping Ratio:
%f \n', ...
        dampedNatFreq,settling_time,dampRatio)

% ---------------------------------------------------------%
% ---------------------------------------------------------
function [rise_time ss_value settling_time
P_over_shoot]=resp_stats(tout,yout)
%
% Steady state value
%
ss_value=yout(end);
%
% Finding the rise time
%
index1=find(yout>=.1*ss_value);
index2=find(yout>=.9*ss_value);
rise_time=tout(index2(1))-tout(index1(1));
%
% Finding the peak time and amplitude
%
index_peak=find(yout==max(yout));
Peak_time=tout(index_peak);
Peak_amplitude=max(yout);
%
% Finding the percent over shoot and settling time
%
P_over_shoot=((Peak_amplitude-ss_value)/ss_value)*100
index_settling_time=find(yout>1.02*ss_value | yout<.98*ss_value);
settling_time=tout(index_settling_time(end));
```

References

[1] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, and B. Wu, Intelligent Fault Diagnosis and Prognosis for Engineering Systems, ISBN: 978-0-471-7299900, 2006.

[2] K. L. Tsui, N. Chen, Q. Zhou, Y. Hai, W. Wang, "Prognostics and Health Management: A Review on Data Driven Approaches," Mathematical Problems in Engineering, vol. 2015, Article ID 793161, 2015. doi:10.1155/2015/793161

[3] J. B. Bowles, "A survey of reliability-prediction procedures for microelectronic devices," *IEEE Transactions on Reliability*, vol. 41, no. 1, pp. 2–12, 1992.

[4] A. Lee, E. Habtour, and S. A. Gadsden, "Proposed health state awareness of helicopter blades using an artificial neural network strategy," *Proceedings of SPIE 9872, SPIE 9872*, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications, Article ID 98720C, 2016. doi:

[5] S. A. Gadsden and T. Kirubarajan, "Development of a variable structure-based fault detection and diagnosis strategy applied to an electromechanical system," *Proceedings of SPIE 10200*, Signal Processing, Sensor/Information Fusion, and Target Recognition XXVI, Article ID 102001E, 2017. doi: 10.1117/12.2262570

[6] R. Ahmed, M. El Sayed, S. A. Gadsden, J. Tjong and S. Habibi, "Automotive Internal-Combustion-Engine Fault Detection and Classification Using Artificial Neural Network Techniques," in *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 21-33, Jan. 2015. doi: 10.1109/TVT.2014.2317736

[7] E. Sobhani-Tehrani, Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach, ISSN 0170-8643, 2009.

[8] M. R. Maurya, P. Praveen K., V. Venkat and R. Rengaswamy, "A framework for on-line trend extraction and fault diagnosis," *Engineering Applications of Artificial Intelligence,* vol. 23, no. 6, pp. 950-960, 2010.

[9] B. Bakshi and G. Stephanopoulos, "Representation of process trends—III. Multi-scale extraction of trends from process data," *Computers and Chemical Engineering,* vol. 18, no. 4, pp. 267-302, 1994.

[10] S. Postalcioglu, E. K. and E. Bolat, "Discrete wavelet analysis based fault detection," *WSEAS Transactions on Systems,* vol. 5, no. 10, pp. 2391-7, October 2006.

[11] R. Isermann, Fault Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance, Springer, 2006

[12] M. Basseville and I. Nikiforov, Detection of Abrupt Changes, Prentice Hall, 1993.

[13] J. P. J. F. D. Chung, "Application of Acoustic Intensity Measurement to Engine Noise Evaluation," SAE Paper, 1979

[14] B. Ray, "Engine Fault Detection Using Wavelet Analysis," Master's Thesis, Windsor, 2007.

[15] E. Leitzinger, "Development of In-process engine defect detection methods using NVH indicators," University of Windsor, Masters Thesis, (M.A.Sc.), Windsor, 2002.

[16] H. Jonuscheit, "Acoustic Tests on Combustion Engines in Production," 2000.

[17] G. Rizzoni and P. S. Min, "Detection of Sensor Failure in automotive Engines," vol. 40, no. 2, 1991.

[18] G. Bloor et al., “An Evolvable Tri-Reasoner IVHM System,’’ Proceedings of *IEEE Aerospace Conference*, Big Sky, MT, 2000.

[19] J. Gray and G. Bloor, ‘‘Application Integration,’’ Boeing’s 3rd Annual Data Exchange Conference, Mesa, AZ, 2000.

[20] G. Karsai and J. Gray, ‘‘Component Generation Technology for Semantic Tool Integration,” *Proceedings of the IEEE Aerospace,* Big Sky, MT, 2000.

[21] S. Guney, “Airbus A420 Series Electrical System,” Aviation Legislation, 24 November 2015. [Online]. Available: http://easamodul10.blogspot.com/2015/11/airbus-a320-series-electrical-system.html. [Accessed 5 May 2017].

[22] D. Hoffman, “Prognostics and Health Management (PHM) / Condition Based Maintenance (CBM),” *IEEE Reliability Society Annual Technology Report*, 2007.

[23] R. Montoya, “Vacuum,” [Digital Image]. Available: https://share-ng.sandia.gov/news/resources/releases/2007/images/vacuum.jpg. [Accessed 5 May 2017].

[24] E. Sobhani-Tehrani, K. Khorasani and S. Tafazoli, "Dynamic Neural Network-based Estimator for Fault Diagnosis in Reaction Wheel Actuator of Satellite Attitude Control System," *In Proceedings of the International Joint Conference on Neural Networks,* 2005.

[25] G. V. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems,* vol. 2, 1989.

[26] X. Wang, N. McDowell, Kruger, Uwe and G. McCullough, "Semi-physical neural network model in detecting engine transient faults using the local approach," *17th World Congress International Federation of Automatic Control, IFAC,* July 2008.

[27] R. K. Yedavalli, "Robust Estimation and Fault Diagnostics for Aircraft Engines with Uncertain Model Data," 2007.

[28] J. Chen, R. Patton and G. Liu, "Detecting incipient sensor faults in flight control systems," in *Proceedings of teh IEEE Conference on Control Applications*, 1994.

[29] H.Schneider and P. Frank, "Observer-based supervision and fault detection in robots using nonlinear and fuzzy logic residual evaluation," *IEEE Transactions on Control Systems Technology,* vol. 4, no. 3, pp. 274-82, May 1996.

[30] W. Li, Yedavalli and K. Rama, "Dynamic threshold method based aircraft engine sensor fault diagnosis," in *Proceedings of the ASME Dynamic Systems and Control Conference*, 2008.

[31] J. Vinsonneau, S. King, P.J. and K. Burnham, "Modelling and observer-based fault detection for an automotive drive train," in *European Control Conference ECC*, 2003.

[32] L. Behera, I. Kar, Intelligent Systems and Control: Principles and Applications, ISBN-13: 978-0-19-806315-5

[33] H. H. Afshari, S. A. Gadsden, S. Habibi, "Gaussian Filters for Parameter and State Estimation: A Review of Theory and Recent Trends," *Signal Processing*, vol. 135, issue C, pp. 218-238, 2017

[34] Y. Bar-Shalom, X. Rong Li and T. Kirubarajan, Estimation with Application to tracking and navigation, New York: John Wiley and Sons, INC, 2004.

[35] A. Gelb, Applied Optimal Estimation, Cambridge, MA: MIT Press, 1974.

[36] T. Kailath, "A view of three decades of linear filterng theory," *IEEE Transactions of Information Theory,* vol. 20, no. 2, pp. 146-181, 1974.

[37] M. S. Grewal and A. P. Andrews, Kalman Filtering: Theory and Practice Using MATLAB, Second ed., New York: John Wiley & Sons, INC, 2001.

[38] S. Haykin, Adaptive Filter Theory, Third ed., Upper Saddle River: Prentic-Hall, 1996.

[39] N. Wiener, Extrapolation, Interpolation, and Smoothing of Stationary Time Series, Cambridge, Massachusetts: MIT Press, 1964.

[40] A. Kolmogorov, Interpolation and Extrapolation of Stationary Random Sequences, Dordrecht: Kluwer Academic Publishers, 1941.

[41] H. W. Sorenson, "Least-Squares Estimation: From Gauss to Kalman," *IEEE Spectrum,* pp. 63-68, 1970.

[42] I. Arasaratnam and S. Haykin, "Squre-root quadrature Kalman filtering," *IEEE Transactions on Signal Processing,* vol. 95, no. 6, pp. 953-977, 2008

[43] I. Arasaratnam, S. Haykin and R. J. Elliott, "Discete-time nonlinear filtering algorithms using gauss-hermite quadrature," *Proceedings of IEEE,* vol. 95, no. 5, pp. 953-977, 2007.

[44] I. Arasaratnam, S. Haykin and R. J. Elliott, "Discete-time nonlinear filtering algorithms using gauss-hermite quadrature," *Proceedings of IEEE,* vol. 95, no. 5, pp. 953-977, 2007.

[45] R. Chen and J. Liu, "Mixture Kalman Filters," *Journal of the Royal Statistical Society, Series B,* vol. 62, no. 3, pp. 493-508, 2000.

[46] R. Chen and J. Liu, "Mixture Kalman Filters," *Journal of the Royal Statistical Society, Series B,* vol. 62, no. 3, pp. 493-508, 2000.

[47] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control,* vol. 54, no. 6, pp. 1254-1269, 2009.

[48] S. A. Gadsden, T. Kirubarajan, "Development of a Variable Structure-Based Fault Detection and Diagnosis Strategy Applied to an Electromechanical System," *SPIE Proceeding 10200, Signal Processing, Sensor/Information fusion, and Target Recognition XXVI,* May 2017. doi:10.1117/12.2262570

[49] B. Ristic, S. Arulampalam and N. Gordon, Beyond the Kalman Filter: Particle Filters for Tracking Applications, Boston: Artech House, 2004.

[50] S. Haykin, Kalman Filtering and Neural Networks, New York: John Wiley and Sons, Inc., 2001.
[51] S. A. Gadsden, "Smooth Variable Structure Filtering: Theory and Applications," McMaster University, Hamilton, Ontario, 2011.
[52] S. A. Gadsden, M. Al-Shabi and S. R. Habibi, "Estimation Strategies for the Condition Monitoring of a Battery System in a Hybrid Electric Vehicle," *ISRN Signal Processing,* 2011
[53] A. Gelb, Applied Optimal Estimation, Cambridge, MA: MIT Press, 1974.
[54] D. Simon, Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches, Wiley-Interscience, 2006.
[55] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," 2006.
[56] S. R. Habibi and R. Burton, "The Variable Structure Filter," *Journal of Dynamic Systems, Measurement, and Control (ASME),* vol. 125, pp. 287-293, September 2003.
[57] S. R. Habibi and R. Burton, "Parameter Identification for a High Performance Hydrostatic Actuation System using the Variable Structure Filter Concept," *ASME Journal of Dynamic Systems, Measurement, and Control,* 2007.
[58] S. A. Gadsden, Y. Song and S. R. Habibi, "Novel Model-Based Estimators for the Purposes of Fault Detection and Diagnosis," *IEEE/ASME Transactions on Mechatronics,* vol. 18, no. 4, 2013.
[59] S. A. Gadsden, M. Al-Shabi, I. Arasaratnam and S. R. Habibi, "Combined Cubature Kalman and Smooth Variable Structure Filtering: A Robust Estimation Strategy," *Signal Processing,* vol. 96, no. B, pp. 290-299, 2014.
[60] S. A. Gadsden and A. S. Lee, "Advances of the Smooth Variable Structure Filter: Square-Root and Two-Pass Formulations," *Journal of Applied Remote Sensing,* vol. 11, no. 1, pp. 1-19, 2017.
[61] S. A. Gadsden, S. R. Habibi and T. Kirubarajan, "Kalman and Smooth Variable Structure Filters for Robust Estimation," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 50, no. 2, pp. 1038-1050, 2014.
[62] S. R. Habibi, "The Smooth Variable Structure Filter," *Proceedings of the IEEE,* vol. 95, no. 5, pp. 1026-1059, 2007.
[63] A. S. Lee, E. Habtour, and S. A. Gadsden, *Proposed Health State Awareness of Helicopter Blades using an Artificial Neural Network Strategy,* 2016 SPIE Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications, Baltimore, Maryland, 2016.
[64] E. Sobhani-Tehrani, Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach, ISSN 0170-8643, 2009.
[65] R. Isermann, Fault Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance, Springer, 2006.
[66] M. Basseville and I. Nikiforov, Detection of Abrupt Changes, Prentice Hall, 1993
[67] M. R. Maurya, P. Praveen K., V. Venkat and R. Rengaswamy, "A framework for on-line trend extraction and fault diagnosis," *Engineering Applications of Artificial Intelligence,* vol. 23, no. 6, pp. 950-960, 2010.

[68] M. Amura, L. Aiello, M. Colavita, F. De Paolis and M. Bernabei, "Failure of a helicopter main rotor blade," *Procedia Materials Science,* vol. 3, pp. 726-731, 2014.

[69] C. Abdallah, G. L. Heileman and D. Docampo, "Neural networks in fault detection: A case study," in *Proceedings of the ASME American Control Conference,* 1997.

[70] D. Pomerleau, Neural Network Perception for Mobile Robot Guidance, Boston: Kluwer, 1993.

[71] E. H. M. a. H. M. Patuwo, "Two-Group Classification Using Neural Netoworks," *Decision Sciences ,* vol. 24, pp. 825-845., 1993.

[72] B. Warner and M. Misra, "Understanding Neural Networks as Statistical Tools," *The American Statistician,* vol. 50, pp. 284-293, 1996.

[73] R. A. Teixeira, A. De Baraga and B. De Menezes, "Control of a Robotic Manipulator Using Artificial Neural Networks with On-line Adaptation," *Journal of Neural Processing,* vol. 12, no. 1, Aug. 2000.

[74] R. Ahmed, M. El Sayed, S. A. Gadsden, S. R. Habibi and J. Tjong, "Artificial neural network training utilizing the smooth variable estimation strategy," *Neural Computing and Applications,* 2015.

[75] R. Ahmed, M. El Sayed, S. A. Gadsden, S. R. Habibi and J. Tjong, "Automotive internal combustion engine fault detection and classification using artificial neural network techniques," *IEEE Transactions on Vehicular Technology,* vol. 64, no. 1, 2015.

[76] Y. Iiguni, H. Sakai and H. Tokumaru, "A Real-Time Learning Algorithm for a Multilayered Neural Network Based on the Extended Kalman Filter," *IEEE Transaction on Signal Processing,* vol. 40, no. 4, April 1992.

[77] S. A. Gadsden, M. El Sayed and S. R. Habibi, "Derivation of an Optimal Boundary Layer Width for the Smooth Variable Structure Filter," in *American Control Conference,* San Francisco, CA, USA, 2011.

[78] H. H. Afshari, S. A. Gadsden and S. R. Habibi, "Robust Fault Diagnosis of an Electro-Hydrostatic Actuator using the Novel Optimal Second-Order SVSF and IMM Strategy," *International Journal of Fluid Power,* vol. 15, no. 3, pp. 181-196, 2014.

[79] M. A. El Sayed, "Multiple Inner-Loop Control of an Electrohydrostatic Actuator," McMaster University, Hamilton, Ontario, 201