JET ENGINE PROGNOSIS USING DYNAMIC NEURAL NETWORKS

Saba Kiakojoori

A thesis

in

The Department

of

Electrical and computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science (Electrical and Computer Engineering)

Concordia University

Montréal, Québec, Canada

January 2014

Concordia University

School of Graduate Studies

This is to certify that the thesis prepared

By: **Saba Kiakojoori**

Entitled: **Jet Engine Prognosis Using Dynamic Neural Networks**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. M. Z. Kabir (Chair)

_____ Dr. A. Bagchi(BCEE) (Examiner)

_____ Dr. S. Williamson (Examiner)

_____ Dr. K. Khorasani (Supervisor)

Approved by _____

Dr.W. E. Lynch,Chair, Department of Electrical and Computer Engineering

____ 2014 _____

Dr. C. W. Trueman Interim Dean,Faculty of Engineering and Computer Science

Abstract

Jet Engine Prognosis Using Dynamic Neural Networks

Saba Kiakojoori

Jet engine related costs and the need for high performance reliability have resulted in considerable interest in advanced health and condition-based maintenance techniques. This thesis attempts to design fault prognosis schemes for aircraft jet engine using intelligent-based methodologies to ensure flight safety and performance. Two different artificial neural networks namely, non-linear autoregressive neural network with exogenous input (NARX) and the Elman neural network are introduced for this purpose. The NARX neural network is constructed by using a tapped-delay line from the inputs and delayed connections from the output layer to the input layer to achieve a dynamic input-output map. Consequently, the current output becomes dependent on the delayed inputs and outputs. On the other hand, the Elman neural network uses the previous values of the hidden layer neurons to build memory in the system.

Various degradations may occur in the engine resulting in changes in its components performance. Two main degradations, namely compressor fouling and turbine erosion are modelled under various degradation conditions. The proposed dynamic neural networks are developed and applied to capture the dynamics of these degradations in the jet engine. The health condition of the engine is then predicted subject to occurrence of these deteriorations.

In both proposed approaches, various scenarios are considered and extensive simulations are conducted. For each of the scenarios, several neural networks are trained and their performances in predicting multi-flights ahead turbine output temperature are evaluated. The difference between each network output and the measured jet engine output are compared and the best neural network architecture is obtained. The most suitable neural network for prediction is selected by using normalized Bayesian information criterion model selection. Simulation results presented, demonstrate and illustrate the effective performance of the proposed neural network-based prediction and prognosis strategies.

Acknowledgments

I would like to express my gratitude to my supervisor Professor Khashayar Khorasani for his sincere encouragement, guidance, and support during my Masters studies. Completion of my masters would not have been possible without his ideas and suggestions. He was always accessible with his vast knowledge. I appreciate all his contributions of time and ideas.

My special thanks are to my committee members, Dr. Kabir, Dr. Bagchi, and Dr. Williamson for devoting their valuable time in guiding and evaluating my work.

I could never thank my amazing aunt, Mrs. Simin Najafi, and her lovely family, especially Mr. Jamil Mohsenin who provided me a warm environment and memorable moments in Montreal during the past 2 years.

Last, but by no means least, my special gratitude goes to my family for their unflagging love and support throughout my life. For my lovely mother whose love, faith and support are always giving confidence and power to explore, grow and develop. For my father who motivates me to work hard by his endeavour and ambition, and for my brother, Nima, who always encourages me.

Contents

List of Figures

List of Tables

List of Abbreviations and Symbols

A	Area, m^2
ANN	Artificial Neural Network
amb	Ambient
C	Compressor
C_P	Compressor pressure
C_T	Compressor temperature
CBM	Condition-based maintenance
CC	Combustion chamber
c_p	Specific heat at constant pressure, $\frac{J}{Kg.K}$
c_v	Specific heat at constant volume, $\frac{J}{Kg.K}$
d	Diffuser
DNN	Dynamic Neural Network
E	Rotor energy, J
EI	Erosion Index
EPR	Engine's Pressure Ratio
FOD	Foreign Object Damage
f	Fuel
FI	Fouling Index
G	Fuel valve gain
GSP	Gas turbine simulation program
H_u	Fuel specific heat,$\frac{J}{Kg}$
i	Stage input
J	Rotor moment of inertia, $\frac{Kg}{m^2}$
LGPA	Linear Gas Path Analysis
Mi	Mixer
M	Mach number
MLP	Multilayer Perceptron
MSE	Mean Squared Error
mech	Mechanical
$\dot{m}$	Mass flow rate, $\frac{Kg}{s}$

N	Rotational speed, rpm
n	Nozzle
NARX	Non-linear Autoregressive Neural Network with Exogenous Input
NMSE	Normalized Mean Squared Error
NBIC	Normalized Bayesian Information Criterion
n_i	Nozzle input
n_o	Nozzle output
o	Stage output
PLA	Power Level Angle
R	Reynold′s number
RMSE	Root Mean Squared Error
T	Turbine
τ	Time constant
T_P	Turbine pressure
T_T	Turbine temperature
u_{fd}	Fuel demand
V	Volume, m^3
W	Mechanical Power
W_f	Fuel flow rate
β	Bypass ratio
γ	Heat capacity ratio
η	Efficiency
π	Pressure ratio

Chapter 1

Introduction

Safety, economy and performance of the aircraft operation is highly dependent on its engines [1]. Jet engines diagnosis and prognosis have been a matter of interest in the recent years due to the increasing demand on reliable operations of these systems. In general, fault detection and diagnosis is a technique to understand whether or not any faults are present in the engine, and determine their locations and severity in its components. Fault prognosis is the ability to predict the future health of the component of the system in a fixed time horizon or its time to failure [2]. Fault diagnosis and prognosis results in performing important condition-based maintenance decisions to reduce maintenance costs due to unnecessary replacements of components or shut downs. In an aerospace industry, jet engine related costs involve a large portion of the operating cost of an aircraft, so fault diagnosis and prognosis allows one to avoid high costs of a failure or overhaul of the system. The overall goal of prognosis is to improve reliability, safety and availability.

Fault prognosis is primarily divided into two main categories namely, model-based

and data-based approaches. Model-based approaches rely on mathematical and physical model of the system while data-based approaches are achieved mostly from historical or real time data from the system measurements to predict the future health of the component. Since there is generally no accurate access to the mathematical equations of an engine, developing model-based approaches would be a challenging task. Data driven methods use real data to approximate the degradation of the components and predict the future behaviour of the system. Moreover, the inherent non-linearity of the jet engine performance makes the need for the application of an alternative computational technique instead of model-based approaches. During the past few years, Artificial neural networks (ANN) which relies on the real-time data from the system components is mostly used as a tool for prognosis [3]. The interest towards neural networks in fault prognosis is due to their ability in modelling non-linearities and complexities. The jet engine is a highly non-linear dynamical system, so in order to model time delays associated with the dynamics of the system; a dynamic neural network is required to learn the dynamics of the aircraft engine.

1.1 Literature Review

1.1.1 Health Monitoring

Modern systems require high precision and reliable performance due to the criticality and complexity. Systems are operating under certain stress or load in the real environment, so no matter how good the product design is, they deteriorate over time which reduces the reliability of the system during its useful life. System health monitoring is an efficient way to assure high reliability of the system. It is a set of activities to maintain the system in its operable condition which may be limited to

the current state of the component or prediction of future operating states [4].

Health monitoring consists of one or more maintenance strategies applied for the purpose of reducing operating costs while at the same time assuring system optimisation, safety, and achieving the highest possible production rate [5]. It is important to balance between reliability and cost. If there is little maintenance, the cost of the system failures will increase. On the other hand, if the maintenance is done too often, the reliability may improve but the cost of maintenance will increase too [6].

Traditionally, maintenance is achieved only at breakdowns. Thus, no analysis or planning is required, which in turn results to unscheduled downtime [7]. Unplanned or run to failure maintenance is practical in small industries with limited maintenance resources [8]. However, in applications such as aircraft engines, reactive maintenance causes critical problems; failure of a component may occur at an inconvenient time or place or it can cause damage to other parts of the system [9].

Another maintenance technique is time-based preventive maintenance, which sets a periodic interval to perform maintenance without considering the health status of the system [5]. This strategy can provide relatively higher system performance. However, this method is quite inefficient for instantaneous failures. Moreover, in pre-defined maintenance activities, the system may be overhauled when they are still in a good health condition. This is money and time consuming process due to the frequent replacements of the expensive components before the end of their lives since engineering components do not fail at periodic intervals. It is also important to determine the maintenance interval to reduce the frequency of undesirable results of system interruptions. Age-related, operating usage or failure distribution is used as

ways to calculate these time intervals. However, Luo *et al.* [10] have stated that the critical system failures are not only based on the time of the system operation. As a result, in the past ten years, many utilities replace their time-based maintenance activities with other efficient programs based on the need of the system to fullfill their needs for the availability and safety of their systems [6].

In order to reduce both maintenance and repair costs and probability of failure, condition-based maintenance (CBM) has been introduced as an efficient way to increase production cycle for modern aircraft which is based on the current health, operating and maintenance history. Variables such as vibration, temperature and acoustic can be used to collect information about the performance of the system [11]. This maintenance method consists of three key steps as shown in Figure 1.1.

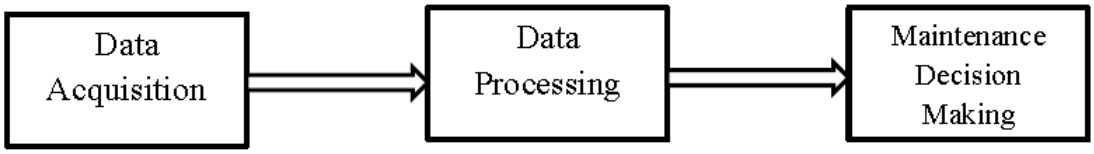

Figure 1.1: Three steps in a CBM program [12].

Data acquisition is the first step where data is collected from the system in different time steps and then they are processed using various methods. In the final step based on the results available one can decide whether it is the time for maintenance actions or not.

This method can significantly reduce maintenance cost and time by reducing unnecessary periodic maintenance operations based on the information collected for health monitoring. A classification for different maintenance approaches is shown in Figure 1.2. Two important aspects in CBM contain diagnosis and prognosis [7].

Reliable diagnostics and prognostics are critical in CBM.

There are different methods for gas turbine condition monitoring including performance analysis, oil analysis, visual inspection, borescope inspection, X-ray checks, vibration monitornig, noise monitoring, turbine exit spread monitoring, etc. [13]. One of the most powerful methods is performance analysis where the information of the degradation severity is obtained based on the gas path parameters [14].

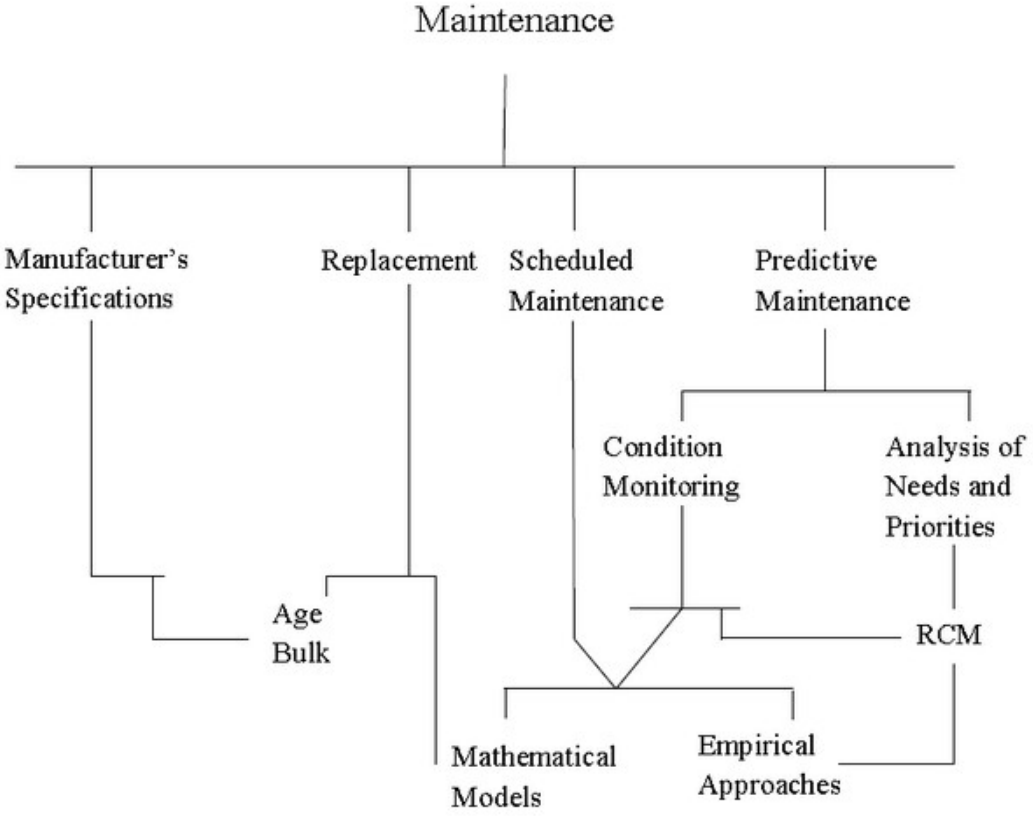

Figure 1.2: Summary of maintenance approaches [6].

1.1.2 Gas Path Analysis (GPA)

Gas turbine performance degrades during operation due to the deterioration of the gas path components [14]. Compressor fouling, foreign object damage (FOD), blade

erosion and corrosion, worn seals and blade tip clearance increase due to wearing are common causes of degradation that may happen in an engine. These degradations can then result in the change in the thermodynamic performance of the engine. The condition of the components can be represented by a set of independent performance parameters. Component efficiencies and flow capacities are mostly used as performance parameters in the literature. These variables are not directly measurable, and they are thermodynamically correlated with engine parameters such as engine rotational speeds, temperatures, pressures, fuel flows, etc. [15]. With the knowledge of these observable measurements, one can determine how an engine performance differs from its healthy state. The most popular diagnostic method known as the gas path analysis (GPA) utilizes this characteristic which was introduced by Urban in 1970s [16] and which is then followed by different derivatives such as optimal estimation based methods.

Linear gas path analysis (LGPA) is based on the linearization of the following equation with respect to X

$Y = F(P, X)$

where P is an input vector such as ambinet pressure or temperature or other environmental measurements and power setting parameter, and X is the component independent variable such as efficiencies and mass flow rates.

Figure 1.3 shows the concept of this method which shows a link between faults, performance parameters and the measured variables [15]. The non-linearity in an engine model is analyzed by other researchers using this method [17, 18]. This method is used to detect only faults which affect measurable variables. In other words, degradations such as blade cracks or sudden failures such as fracture can not be detected by using GPA because they do not have a direct effect on measurable variables.

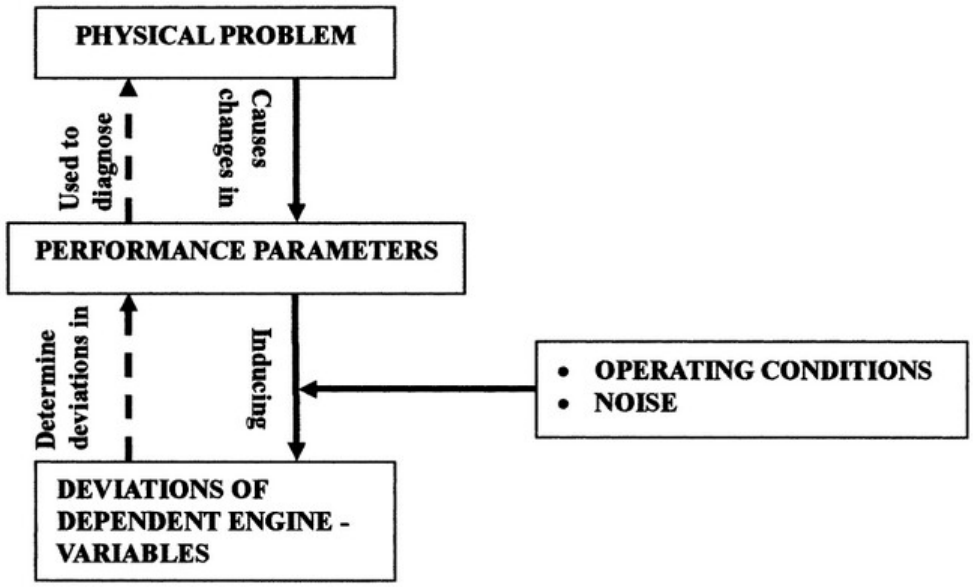

Figure 1.3: Conceptual framework of GPA [15].

1.1.3 Diagnostics

Fault detection and diagnosis (FDD) is a technique to detect faults based on the deviations between component parameters and their nominal values, and to determine their locations and significance in a system being monitored [19]. Faults are generally regarded as any kind of malfunction in a system which leads to system instability or unacceptable performance degradation. Faults can occur in different parts of the engine namely; actuators, sensors or system components. FDD contains fault detection, isolation and fault identification [19]. Fault detection is a decision whether something is going wrong in the system or not. It alerts the occurrence of the fault, while isolation locates the component which is faulty and its type, and finally identification is the task to determine the nature and the magnitude of the fault which is detected. In the literature, fault isolation and identification are usually referred to as diagnosis. Diagnosis is the art or act of identifying a condition from its signs or symptoms [5]. The overall goals of diagnosis are to correctly isolate and identify physical faults that

consist of variety of problems or combinations of anomalies. A schematic view of the gas turbine fault diagnosis is shown in Figure 1.4.

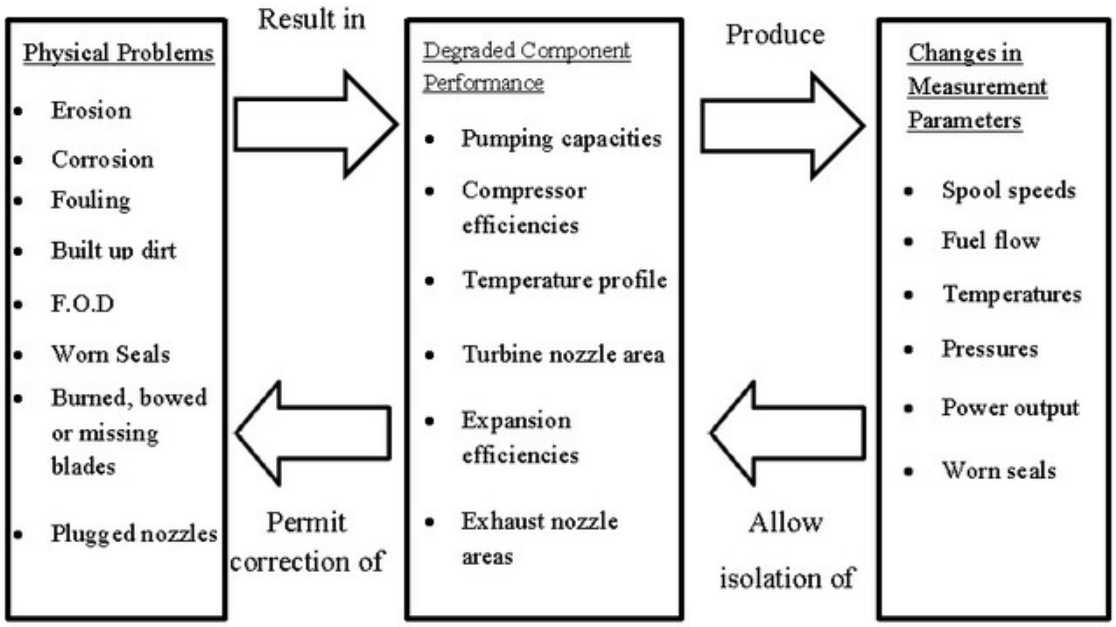

Figure 1.4: Gas turbine fault diagnosis approaches [16].

Traditionally, extra hardware, known as hardware redundancy, was used to achieve fault diagnosis. However, this method is not practical because an increase in the number of sensors contributes to an increase in cost, weight and complexity [20]. Consequently, analytical redundancy which eliminates the need for extra hardware was introduced in early 1970s by Beard [21]. The first step in an analytical redundancy FDD is to generate residual which is the difference between healthy model output and the actual output. This residual would be used to decide whether or not faults have occured by methods such as a threshold test on the value or other complex statistical methods such as likelihood ratio testing [22]. The residual can then be processed using signal processing methods or frequency analysis for fault isolation and identification steps. A comparison between hardware and analytical redundancy is shown in Figure

1.5.

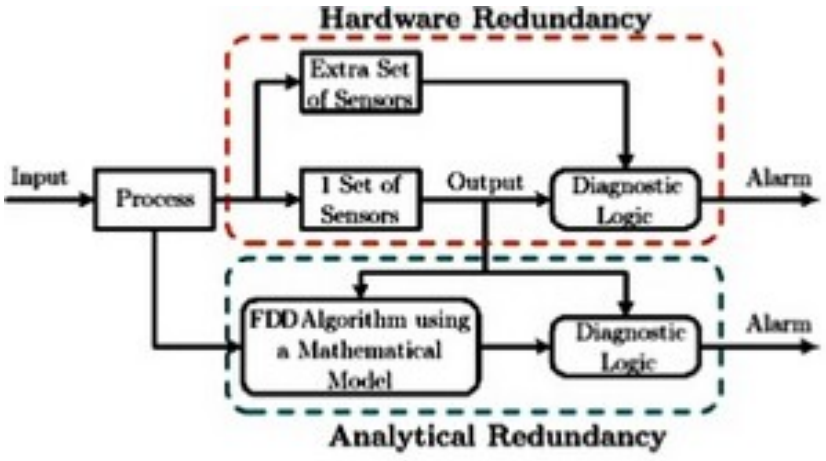

Figure 1.5: Analytical versus Hardware redundancy based FDD [23].

Fault detection of jet engines has been reported in the literature using methods such as model-based, data-driven, and expert system-based approaches, or a hybrid methods. A review of these methods was performed by Marinai *et al.* [24]. Denny has first used neural networks in gas turbine diagnosis applications in 1965 [25]. These data-based methods were used extensively since 1989. Expert systems were first introduced to gas turbine applications in the early 1980's. These methods are still under development. One of the major developments is the use of rule-based fuzzy expert systems in the diagnosis of gas turbine engines.

Model-based fault diagnosis of jet engine based on a bank of linear Kalman filters was proposed in [26]. Yedavalli *et al.* [27] present a non-linear dynamical model of a two-spool turbofan engine in MATLAB/Simulink and diagnose sensor faults using Kalman filter. Recently, dynamic neural networks have been proposed for fault diagnosis of non-linear systems. Valdes *et al.* [28] have developed a multilayer perceptron network embedded with dynamic neurons for fault detection and isolation of thrusters in the formation flight of satellites and fault detection had been done

using dynamic neural networks by [29] for gas turbine engines. The authors in [30] implement artificial neural networks for a high bypass ratio military turbofan engine, and faults are detected in an aircraft engine with noisy sensor measurements with the use of data-driven approaches [31]. Statistical analysis is also used to estimate faults in an aircraft jet engine. This algorithm is then validated with the NASA C-MAPSS model of commercial aircraft engine [32]. A regression-based approach was used for detecting anomalies in aircraft performance during cruise flight [33]. Finally, Nan *et al.* [34] have used the advantage of expert-based systems as a diagnose tool for two different case studies; simulated data of a micro steam power unit and data performed in a real process environment.

Fault isolation has been done in the literature using different methods. Multiple model approach was implemented for the fault isolation of sugar evaporation process in [35]. Al-Zyoud *et al.* [36] isolated faults using a self-organizing map network followed by a linear vector quantization network and fault isolation of a satellites actuator is accomplished by using recurrent neural network by Li *et al.* [37].

1.1.4 Prognostics

Machines usually go through degradation processes before they fail completely [38]. Prognosis is the process of predicting the future state of the system based on the current state of the component and past operation profile, which is one of the major challenges for control engineers [39]. The current measures of a variable along with its past observations is used to develop the model to describe the relationship between variables. This model is then used to extrapolate the variable into future. The accuracy of all prognosis approaches is highly dependent on the accuracy of the

measurements. Sensor noise, disturbance, instrument degradation and human errors are among reasons which cause uncertainty in measurement values [18].

Traditionally, prognosis of machine components were calculated off-line based on the statistical models of the mechanical properties of the material, operating conditions and major disruptions such as an emergency shut-down of the component from its full load. Nevertheless, with the progress in the area of prognosis, on-line degradation rate will assist one in making decisions about the health and safety of the component [40].

Prognosis is the ability to predict the time to failure or remaining useful life of a subsystem which requires precise models to predict future machine health states. However, for critical systems such as aircraft, it is used to warn before the machine reaches a predetermined threshold, and therefore avoid the catastrophic failures or aerial casualties. Generally speaking, prognosis is used in order to provide enough time for maintenance planning and consequently, reduce unnecessary maintenance actions [41].

In order to reach the safety goal through prognosis, current state of the component and degradation progression should be described accurately [42]. "prognosis is the most difficult of maintenance tasks; one must be able to diagnose faults before one can perform prognostics" [43]. Therefore, the progress which had been made in fault detection and diagnosis is more extensive than prognosis. Different condition parameters can be used in prognosis, namely vibration signature, oil analysis, acoustic data, temperature, pressure, moisture, etc. [7, 38].

Prognosis can generally be applied in three different approaches based on the way the knowledge about the system is utilized. The hierarchy of different prognosis

approaches based on their applicability and accuracy is shown in Figure 1.6.

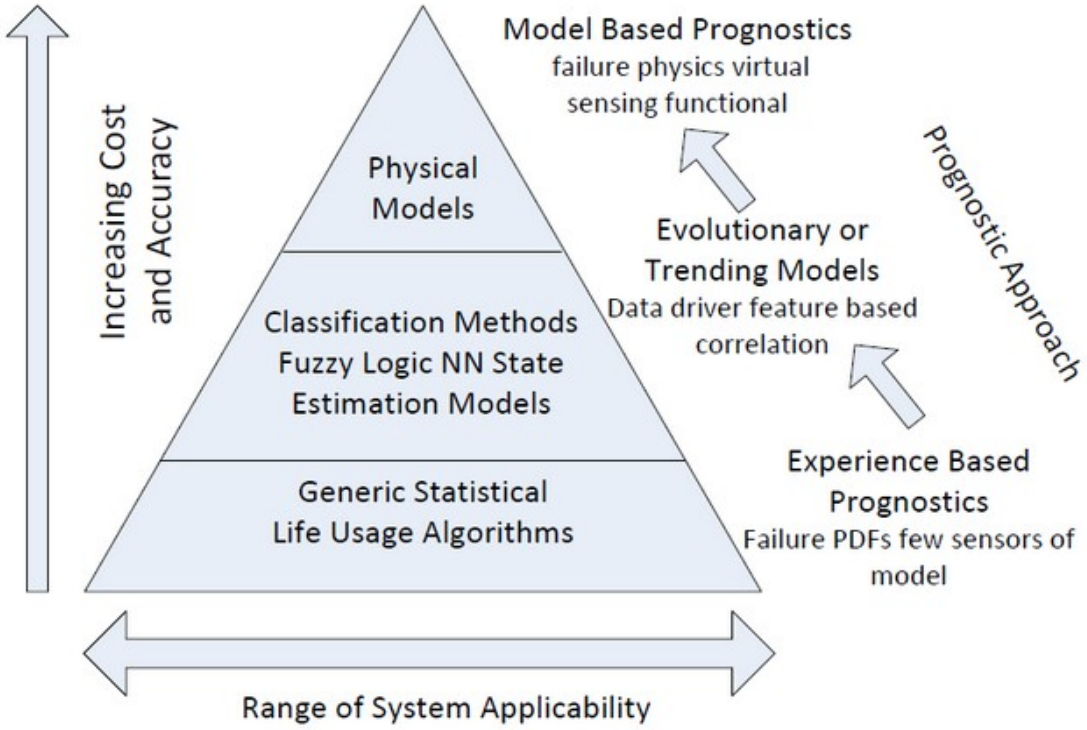

Figure 1.6: Three prognosis methods [44].

The first one is the mathematical model-based framework where *a priori* knowledge of the system is represented by the system mathematical model derived by using physical fundamentals. Thus, it requires a specific mechanistic knowledge of the equipment [7]. Orchard and Vachtsevanos [45] used a particle filter-based approach to analyse the growth in a crack on a turbine engine blade. They reported that particle filtering is useful in dealing with complex dynamic systems and/or non-Gaussian problems such as engines. Ray and Tangirala [40] used a non-linear stochastic model of fatigue crack dynamics in mechanical structures and Abbas *et al.* [46] estimated the remaining useful life of electrical power generation and storage (EPGS) such as

battery and electrical loads using a mechanistic model. Hidden semi-Markov model (HSMM) algorithm was developed and validated on a real-world helicopter rotor track to predict the time to the next required rotor track maintenance [47]. Watson *et al.* [48] predicted the remaining useful life of highly dynamic clutch systems using physical model of the system. Chelidze tracked battery degradation of a vibrating beam system to estimate the remaining useful life for battery discharge process [49].

The performance of model-based prognosis mainly relies on the ability of the dynamic model to follow the trend of the studying process [50]. The main advantage of this approach is that it can incorporate physical understanding of the system under study [51].

While model-based approaches provide acceptable results in terms of precision and accuracy, it is usually quite challenging to find an exact mathematical model of the system due to the existence of uncertainties, noise and disturbances to mimic the real life of the system. Moreover, increasing the system non-linear complexities and model due to the limitations of the analytical model-based technique, decrease their reliability for health monitoring.

The second framework is the data-based or computational intelligence-based framework which is based on the nominal and degraded data collected using statistical or artificial intelligence techniques. Thus, these methods are suitable when there are enough data to specify the dynamics of system under monitor. These approaches can be divided into two categories: Artificial intelligence (AI) techniques and statistical ones [51]. Among the AI techniques one can list the following:

- Neural networks (multi-layer perceptron, probabilistic neural networks, learning vector quantization, self-organizing maps, etc.) [51],

- Fuzzy rule-based systems and neuro-fuzzy systems [52],
- Decision trees [53],
- Graphical models (Bayesian networks, hidden Markov models) [54],

and among the statistical techniques we can list:

- Multivariate statistical methods (static and dynamic principle components (PCA)) [51],
- Linear and quadratic discriminant [51],
- Partial least squares [51],
- Canonical variants analysis [51],
- Signal analysis (Auto-Regressive model, etc.) [55],

Data-driven approaches based on the large amount of data that are available online, such as artificial neural networks (ANN), neuro-fuzzy systems (NFs), and support vector regression (SVR) have become the primary prediction tools of complex systems [56]. These methods are preferred when operational aircraft data is available using sophisticated sensors and database software where data represent the systems behaviour being monitored. Various works emphasize on using these methods for prediction problems [57, 58].

Combination of a predetermined level of failure probability and auto regressive moving average (ARMA) model is used to estimate the remaining useful life by Yan *et al.* [59] . Hidden Markov model and proportional intensity model are considered as powerful tools for prediction estimation in [54, 60], and Garga *et al.* [55] used a signal analysis method for industrial gearbox prediction. Stochastic autoregressive

moving average (ARIMA) models were also used in a prognostic system [61].

Among the intelligent-based approaches, artificial neural networks have received a lot of attention due to their promising capabilities in learning the dynamics and input-output relations of a system. There is no need to specify an exact model form with using ANNs. Thissen *et al.* [111] applied three methods, namely, ARMA models, support vector machines (SVM) and Elman networks for prediction of three different data sets. ARMA models are easy and fast to use. However, they have a critical disadvantage which is their linear behaviour that makes it difficult to model non-linear relations. SVMs training are much longer, but they yield a global solution in contrast to other networks where the solution is based on the training.

Different neural network methods have been used in the area of prognosis because of their flexibility in generating suitable models. Vachtsevanos and Wang [58] introduce a prognostics framework based upon concepts of dynamic wavelet neural networks and its practicality is checked via a bearing example. Polynomial neural networks were used as an estimation scheme for the analysis of normal and defective vibration signatures in helicopter transmissions [63]. Huang *et al.* [64] predict the life of ball bearings based on self-organizing map and back propagation neural network methods.

Finally, the third framework includes the expert system-based or fuzzy rule-based approaches, which use an expert knowledge of the system operation and its modes to obtain a qualitative model of the system. In this approach, rules are used to represent the situation under studying. The main advantage of expert systems is that it does not need real data. However, a knowledgeable expert is needed to fix the rules to

follow the structure of the component [2]. Frelicot [65] designed a prognostic adaptive system based on fuzzy pattern recognition and Bonissone [66] used fuzzy logic as a tool for prognosis. This approach in prognosis is not commonly used because of the non-existence of numerical condition of data [41].

There are various researchers who combine one or more of these methods to take advantage of each of them. Zhang [67] combines model-based and computational-intelligent-based methods to increase the precision of the prediction. Remaining useful life of a degraded gear is also predicted using both gear physical model and real-time condition monitoring data [68]. Gao and Joo [104] compare three different time series prediction using neural network and neuro-fuzzy systems and demonstrate that the combination of these two methods outperform previous approaches. In [70] a fuzzy back propagation network is presented to estimate the remaining useful time in induction motors. Brotherton *et al.* [2] combined dynamically linked ellipsoidal basis function (DL-EBF) neural network with rule extractors to the vibration data in gas turbine engine prognostics. Fault detection, diagnosis and prognosis of an aircraft are accomplished in [71] by using both model-based and data-driven approaches. These combinations possibly offer more reliable prognostic results [43].

1.1.5 Artificial Neural Networks

Artificial neural network (ANN) is currently the most commonly used data-driven approach in the prognostics literature. They consist of a layer of input nodes, one or more layers of hidden nodes, a layer of output nodes which are interconnected via weighted links [3]. The number of layers and the way the neurons and layers are connected are dependent on the type of network. An example of a simple neural

network is shown in Figure 1.7 [72]. Artificial neural networks which are based on data-driven approaches do not require a detailed mathematical model of the system. These methods are based on the real-time or historical data which are collected from sensors to track, approximate and forecast the system degradation behaviour [88].

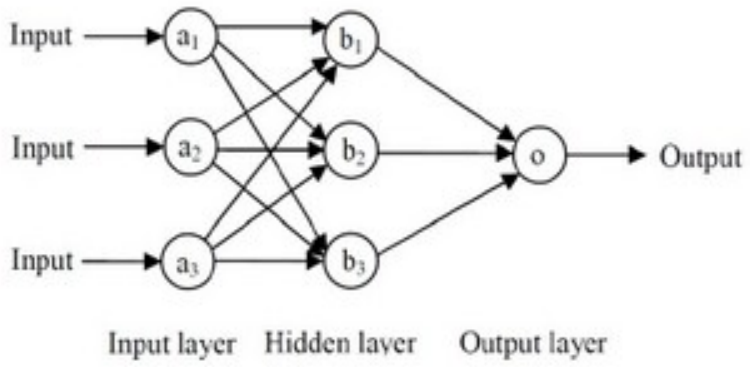

Figure 1.7: Simple example of a neural network [72].

Different studies have shown the merits of ANN such as performing faster than system identification techniques in multivariate prognosis [74] and capturing complex phenomenon without *a priori* knowledge. With the technology of sophisticated sensors and database software, neural networks as a popular fault prognosis method have been extensively studied and discussed in the literature. Recurrent neural networks (RNN) are used in [75] and [76] to trend condition monitoring indices and forecast the next time step. Jianzhong *et al.* [77] demonstrate the concept of multiple layer perceptron (MLP) neural networks to model the remaining useful life of a NASA turbofan engine degradation simulation data set. Artificial neural network is also used for modelling and prediction of complicated time series [78]. In [79], Gebraeel and Lawley develop a modular neural network-based degradation model that utilizes degradation signals to compute residual life of a degraded rolling bearing. Dragomir *et al.* [80] utilized adaptive neuro-fuzzy inference for stabilizing the error

of the prognosis. Recurrent radial basis neural networks have been used by Zemouri for prognosis of non-linear gas ovens [81]. The time to failure of a bearing system using self-organizing neural network is developed in [82].

The strength of these methods is their capability to transform high-dimensional noisy data into lower dimensional information which can be used for prognostic decisions [51]. However, they are highly dependent on the system operational data. Neural networks have been successfully applied to fault prognosis problems due to their capabilities to cope with non-linearity, complexity, uncertainty, noisy or corrupted data. Neural networks are very good modelling tools in learning transformations that map a set of inputs to a set of outputs.

1.1.6 Dynamic Neural Networks

Static neural networks suffer from some drawback; the information flow is in one direction, from input to output, and there is no feedback in the system. Moreover, there is no modelled time delays associated with the dynamics of the system. Consequently, this kind of neural network is not applicable for use in a highly non-linear dynamic system such as gas turbine engine. Dynamic neural networks solve the above mentioned problem. Recently, dynamic neural networks have been developed due to their high capabilities in modelling complicated non-linear dynamical systems. There is an internal or external feedback in the DNN which provide us with the memory of the system and can generate dynamic input-output behaviour. Most of the static neural networks are used for off-line and steady state engine fault prognosis. In contrast, dynamic neural networks have the capability in modelling non-linear dynamic systems [83].

There are mainly two different architectures for DNN feedback; global feedback and local one. The first can be achieved when the network output fed back to the input layer as shown in Figure 1.8 while local feedback is a feedback from hidden layers to the other layers as depicted in Figure 1.9.

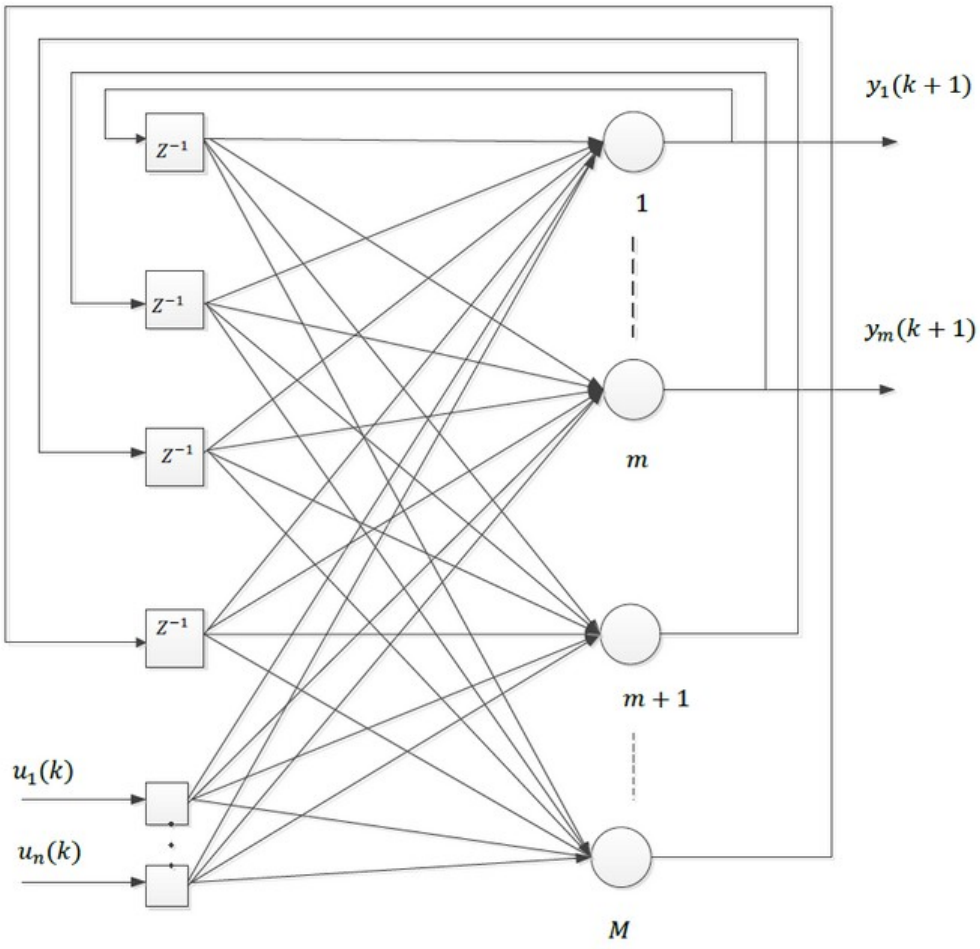

Figure 1.8: Global recurrent network structure [84].

Yazdizadeh *et al.* [85] presented a form of dynamic neural networks where the filter is placed after the activation function of the neuron. A dynamic wavelet neural network is proposed by Vachtsevanos and Wang [58] to predict the failure using vibration data from cracked bearings in industrial chillers. Vibration data from degradations

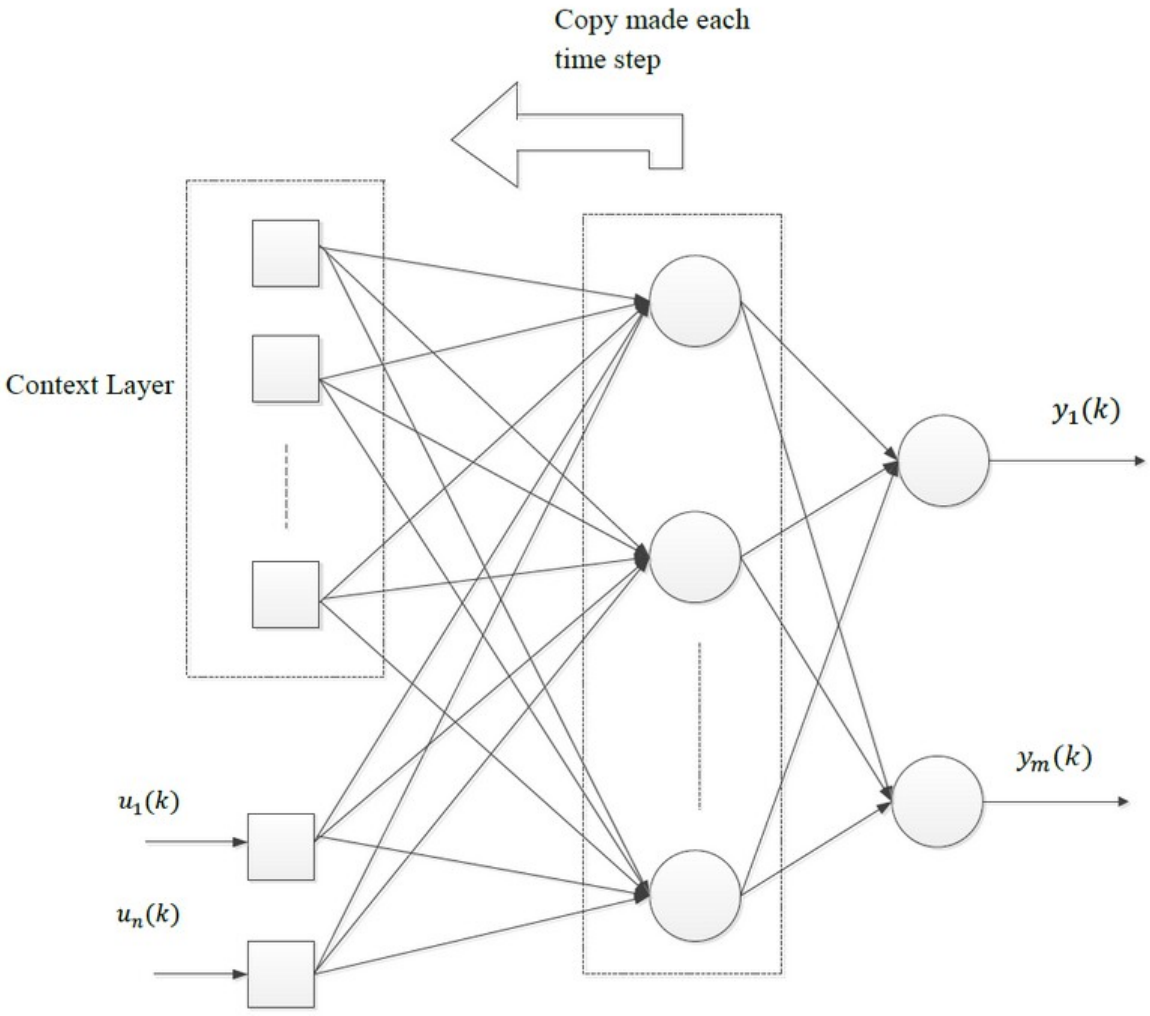

Figure 1.9: Local neural network architecture [84].

in helicopter gear boxes is trained using polynomial neural networks for prognostic considerations [63], and Lee *et al.* [38] employed an Elman neural network for health condition prediction. Wang *et al.* [52] compared the results of the prediction of fault damage trend using two methods of recurrent neural networks and neuro-fuzzy inference systems. They also proved the robustness of ANN approaches in their work [86]. Jordan networks and Buffer networks with sigmoidal output and hidden units are developed in [87] as a tool for time series prediction.

NARX networks which is a kind of dynamic neural networks have been used as a tool for system identification and time series prediction in many applications. NARX neural network is also used to model non-linear systems [88, 89]. Catalytic reforming systems in a petroleum refinery are predicted using NARX [90] and non-linearity in heat exchangers is modelled via this network [91]. A multi-step ahead NARX response time predictor for MySQL database server has been proposed by Amani *et al.* [92]. It is concluded that this architecture is simple, non-linear, and measurements can be obtained without any requirements on changing operating systems. Diaconescu [93] tested the performance of NARX networks in prediction for different time series. Online multi-step ahead forecasting using NARX was implemented in [94]. NARX neural network and Elman network are used to predict storm time in [167]. They concluded that the NARX network shows much better capability than Elman network.

Elman neural network is used as a prognosis method in different areas to study the behaviour of processes in time. Thissen *et al.* [111] compared three different methods; namely support vector machines (SVM), Elman neural network and autoregressive moving average (ARMA) models in time series prediction. Application of the radial basis function (RBF) neural network and Elman network in CBM is presented in [117]. They concluded that if the maintenance strategy considers more attention on the continuously running process to avoid unnecessary shut-downs, Elman network

outperforms RBF. Elman network is used by Yang *et al.* [112] to predict burning zone temperature in the cement rotary kiln calcining process which is a kind of functional equipment for fuel combustion, heat exchange, and chemical reaction.

1.2 Thesis Contributions

Solutions to the problem of jet engine prognosis when degradations occur in the engine are developed for performing condition-based maintenance actions. The contributions of this thesis are detailed as follows:

- Compressor fouling and turbine erosion are modelled in the single spool jet engine by considering the effects of these deteriorations on the thermodynamic parameters of the engine. Concurrent degradations consisting of compressor fouling and turbine erosion are also modelled for different degradation rates.

- Compressor fouling, turbine erosion, and concurrent degradations of these deteriorations are modelled in the GSP software. The generated data are then compared to the data generated from our Simulink model to evaluate the validity and accuracy of the data which are used in this work.

- Different NARX neural network architectures are developed to predict multi-flights ahead turbine temperature. These networks are trained and evaluated by various amount of data points. The statistical error measures such as mean, standard deviation, and RMSE for each network are calculated and the optimal networks are determined for performing temperature predictions.

- Elman network structures are applied to the jet engine degraded data. The networks are trained by various number of training and testing data points and

their performance in multi-flights ahead prediction are compared collectively to determine the best network for engine temperature prediction.

- Normalized Bayesian information criterion is used as a method to compare the results of the prediction for the NARX networks and the Elman neural networks. It has been observed that the Elman neural network has better prediction capabilities in comparison to the NARX neural network.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. In Chapter 2, the background information on the architecture of the two proposed neural networks, namely the NARX and the Elman neural networks as well as their learning rules are briefly reviewed. The non-linear mathematical model of a single spool jet engine and its equations are presented. Degradations in the jet engine are also considered. Compressor fouling and turbine erosion are modelled in our Simulink model. The generated data are then validated and evaluated with the data generated from the GSP software discussed in Chapter 2. The proposed multi-flights ahead turbine temperature prediction by using the NAXR neural network as well as the simulation results for different scenarios and cases are presented in Chapter 3. The results of applying the Elman neural network for the jet engine prediction are presented in Chapter 4. In Chapter 5, our proposed methods are compared together. The concluding remarks followed by the future work are also presented in this chapter.

Chapter 2

Background Information

In this chapter, we present an overview of the background material related to our work. In this thesis we have studied degradation prediction of aircraft jet engines using neural network methodologies. We first describe and introduce non-linear autoregressive neural network with exogenous input (NARX) as an efficient tool to predict non-linear dynamic systems. Next, we describe the Elman neural network which will later be used for engine temperature prediction. The non-linear mathematical model of a single spool jet engine that is used to develop a Simulink model of the system for data generation is briefly described. Common degradations in the jet engine are explained, and 2 important ones, namely compressor fouling and turbine erosion are modelled in our Simulink model. These degradations are also modelled in the gas turbine simulation program (GSP). Finally, the generated data are validated and evaluated with the data generated from the GSP software.

2.1 Non-linear Autoregressive Neural Networks with Exogenous Input (NARX)

Among data-driven methods, artificial neural networks (ANN) use data to capture functional relationships between input and output measurements. They imitate the functional behavoiur of neural systems in the nature. They are dependent on both weights and input-output functions (activation functions) that are specified for the neurons. Common activity functions are the sigmoidal or the tangent hyperbolic functions [95].

One of the most convenient model forms for prediction purposes is the non-linear autoregressive model with exogenous input (NARX) where the current output value is dependent on the lagged inputs and outputs that map through the network non-linear functions [96]. This non-linear function can be described as a feed-forward neural network, polynomial expansion, radial basis functions, wavelets, support vector machines, etc. [98]. This network is an important model of discrete-time non-linear systems which uses global feedback from its output layer [97].

Although recurrent architectures have feedback from hidden neurons, NARX network feedback comes only from the output neurons [102]. Gradient-descent learning in NARX networks is more effective than in other recurrent networks due to the embedded memory of these networks which reduces the sensitivity to long-term dependencies [103]. It was also stated that convergence in these networks is much faster than other networks [104, 105]. Choosing a suitable network and memory for prediction are important issues in NARX networks where the variance of models predictions increases as the number of regressors increase [106].

2.1.1 NARX Structure

NARX networks use a tapped delay line from the input and delayed connections from the output of the last layer to the input layer. There are two different structures for the NARX network [107], as described below:

- Parallel (P) mode in which the estimated output (the network's output represented as $\hat{y}(k)$) is fed back to the input of the feed-forward neural network as part of the standard NARX structure as shown in Figure 2.1, to yield

$$\hat{y}(k+1) = f[\mathbf{y}_P(k); \mathbf{u}(k)] = f[\hat{y}(k), \ldots, \hat{y}(k-d_y+1); u(k), \ldots, u(k-d_u+1)], \tag{2.1.1}$$

  where d_u and d_y are input delays and output delays, respectively, and f is the non-linear mapping function.

  Equation (2.1.1) implies that the network receives the past and the present values of the input as well as the past and the present estimated values of the output as inputs and the next value of the output as the target in the training phase. The trained network is then used to estimate the value of the next step for the unseen data in the testing phase.

  NARX neural network can be trained to predict multi-steps ahead based on equation (2.1.2) where the present and the past observations $u(k)$,...,$u(k-d_u+2)$,$u(k-d_u+1)$ and the present and the past estimated outputs $\hat{y}(k)$,...,$\hat{y}(k-d_y+2)$,$\hat{y}(k-d_y+1)$ are used as input variables, and the output in the n-step ahead as the target value in the training phase, that is

$$\hat{y}(k+n) = g(u(k-d_u+1), u(k-d_u+2), \ldots, u(k), \hat{y}(k-d_y+1), \ldots, \hat{y}(k)) \tag{2.1.2}$$

  where d_u and d_y are input delays and output delays, respectively, and g is the

non-linear mapping function.

As an example, to train the network to predict n=5 steps ahead output, the network is trained using the present and the d_u past values of the input based on the number of input delays and the estimated present and the past d_y values of the output based on the number of output delays as the input variables. The value of the output in the 5 steps ahead is treated as the target in the training phase. When the network is trained, the available data up to the present instant are given to the network to predict the 5-step ahead output value.

- Series-parallel (SP) structure in which the actual output is used instead of feeding back the estimated output as shown in Figure 2.2, to yield

$$\hat{y}(k+1) = f[\mathbf{y}_{SP}(k); \mathbf{u}(k)] = f[y(k), \ldots, y(k-d_y+1); u(k), \ldots, u(k-d_u+1)], \tag{2.1.3}$$

where d_u and d_y are input delays and output delays, respectively, and f is the non-linear mapping function.

Based on equation (2.1.4), the present and the past d_u values of the input variables and the actual present and the past d_y values of the output variables are given as the inputs to the network, while the n-step ahead value of the output is treated as the target of the system in the training phase. When the network is trained, the available data up to the present instant are given to the network to predict the n-step ahead output variable.

$$\hat{y}(k+n) = g(u(k-d_u+1), u(k-d_u+2), \ldots, u(k), y(k-d_y+1), \ldots, y(k)) \tag{2.1.4}$$

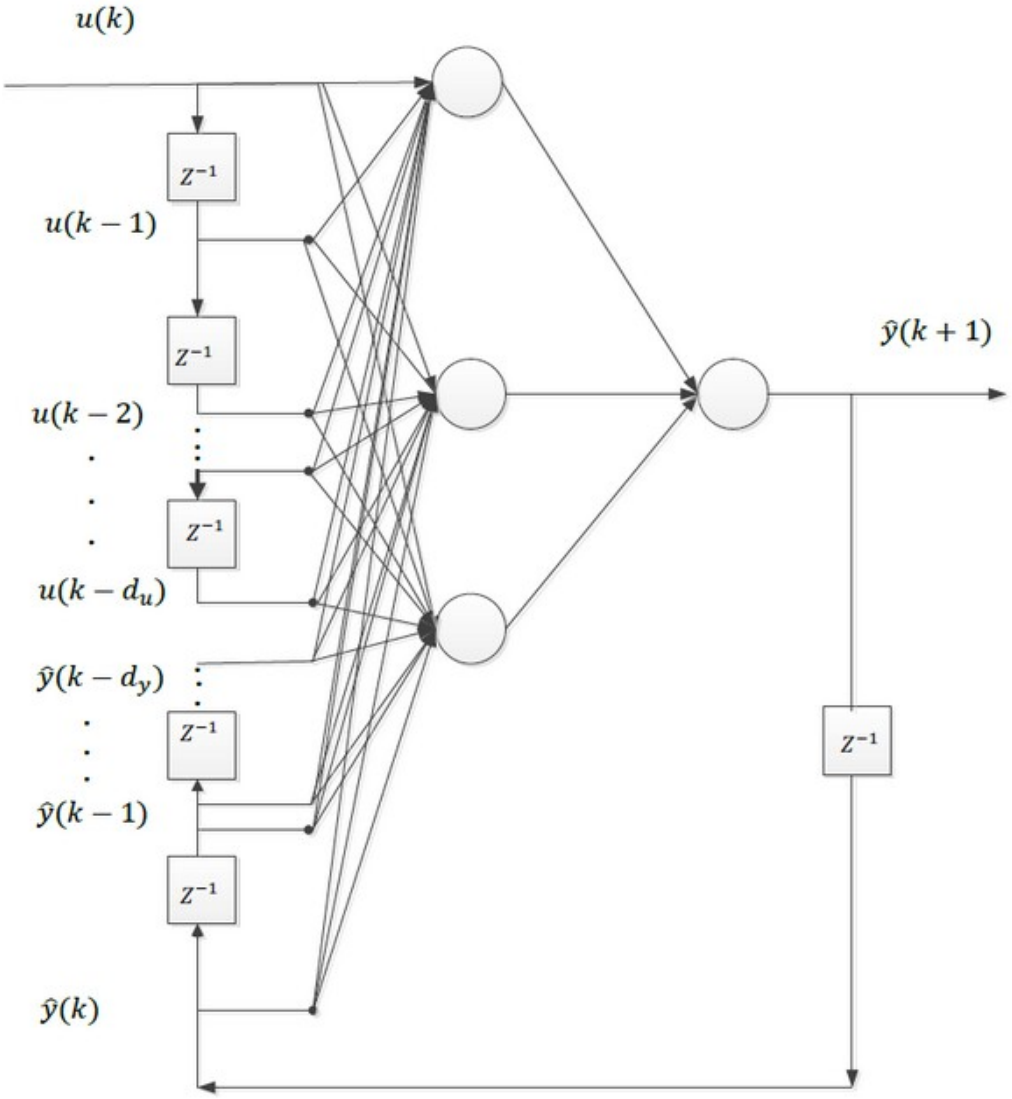

Figure 2.1: Parallel architecture for the NARX network [108].

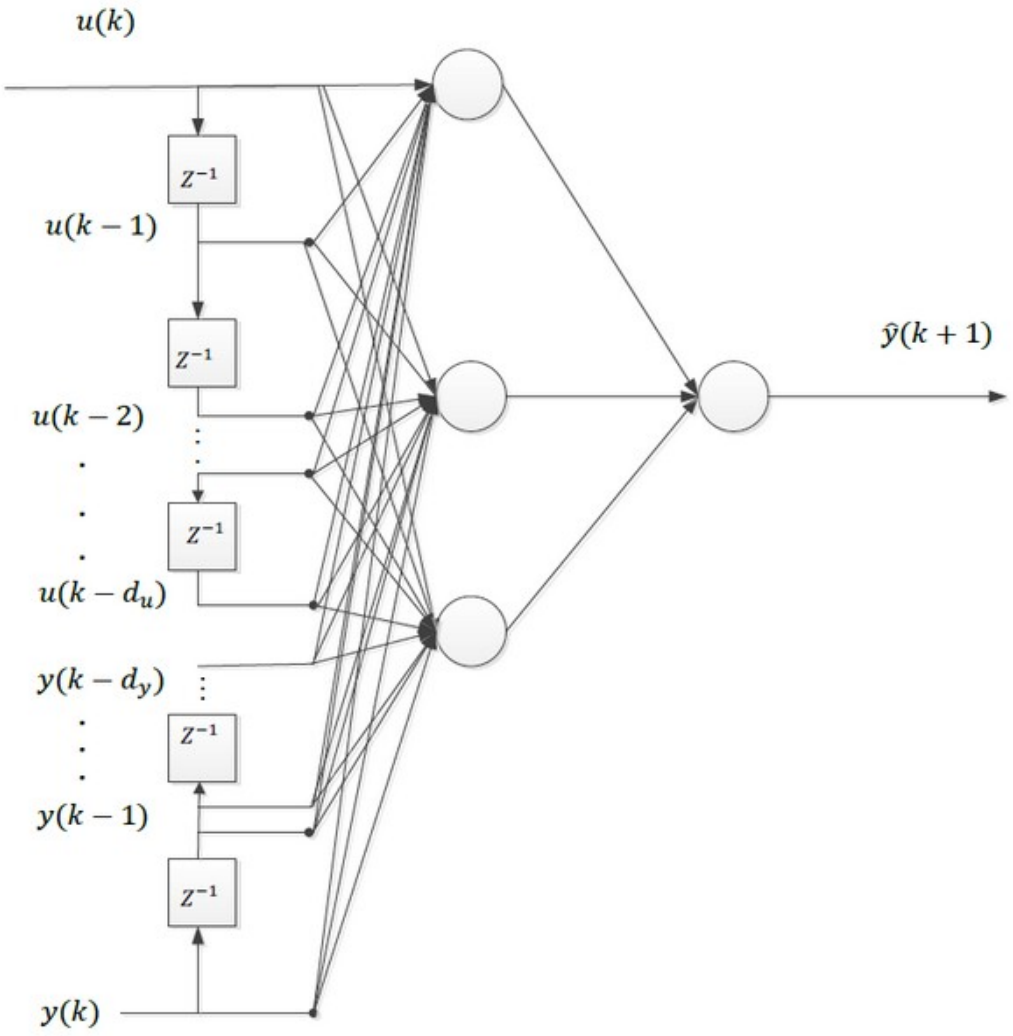

Figure 2.2: Series-parallel architecture for the NARX network [108].

The difference between the NARX network and other recurrent networks is that the feedback comes only from the output layer and not the hidden layers. When the output memory order of the NARX network is zero, a NARX network becomes a time delay neural network (TDNN) which has only a tapped delay line of the input nodes [110].

2.1.2 NARX Network Learning Algorithm

Neural network learning can be seen as a function optimization problem to determine the best network parameters to minimize the network error. Two different approaches are available based on the type of network used.

- Supervised learning, where both input and output pairs are presented to the network during the training process so that the network can adapt its weights in the way to obtain desired output from the input.

- Unsupervised learning, where the neural network is only provided with the input values, and the network adjusts the weights based solely on the input values and the current network output. The training algorithm modifies net weights so that it produces outputs that are consistent, that is, application of two similar inputs produces the same pattern of outputs.

Basic back propagation is one of the most popular learning methods for performing the supervised learning task. Its goal is to adjust the parameters of the network based on a given set of input-output pairs in order that it generalizes well for patterns from outside the training set. The idea of the standard back propagation is widely applied in static contexts and has extensions to dynamical systems. Dynamic networks can be trained using standard optimization methods. However, the gradient and the Jacobians that are required for these methods cannot be computed by using

standard back propagation algorithm [100]. Dynamic back propagation algorithm is required to compute the gradient for dynamical networks which is used in this thesis. The weight and bias updates use the Levenberg-Marquardt optimization [93] which minimizes a combination of squared error of estimated and actual values of output and weights, and then determines the correct combination to minimize a non-linear function. This algorithm has the fastest convergence in networks that contain up to a few hundred weights and it has a stable convergence [93]. It uses both the stability of the steepest descent method and the speed of the Gauss-Newton algorithm in its convergence.

The proper synaptic weights of the network can be obtained by using the batch mode of the backpropagation learning algorithm to find the minimum errors during the training step where the error reaches a stable condition. The cost function or the performance index is minimized by the steepest descent method. The trained network as well as its weights stay fixed to be used in the testing step [94].

The cost function or the performance index (F) which is to be minimized is defined as

$$F = 1/2[\sum_{k=1}^{K}(d_k - y_k)^2] \tag{2.1.5}$$

where d_k is the desired value of the k^{th} output, y_k is the network value of the k^{th} output, and K is the number of the neuron outputs.

Equation (2.1.5) can be written as

$$F = E^T E \tag{2.1.6}$$

where E is the error vector defined as,

$$E = [e_1 \dots e_K]^T \tag{2.1.7}$$

where

$$e_k = d_k - y_k, \quad k = 1, \dots, K \tag{2.1.8}$$

In the training process, the value of the weights in each iteration can be calculated as

$$w(t+1) = w(t) + \Delta w(t) \tag{2.1.9}$$

where

$$\Delta w(t) = (J(t)^T J(t) + \mu(t) I)^{-1} J(t)^T E(t) \tag{2.1.10}$$

and where μ is the learning rate, I is an identity matrix, and J is the Jacobian matrix which is defined as [162]:

$$J = \begin{bmatrix} \frac{\partial e_1}{\partial w_1} & \frac{\partial e_1}{\partial w_2} & \cdots & \frac{\partial e_1}{\partial w_N} \\ \frac{\partial e_2}{\partial w_1} & \frac{\partial e_2}{\partial w_2} & \cdots & \frac{\partial e_2}{\partial w_N} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial e_K}{\partial w_1} & \frac{\partial e_K}{\partial w_2} & \cdots & \frac{\partial e_K}{\partial w_N} \end{bmatrix} \tag{2.1.11}$$

where N is the total number of parameters (weights+biases). For $\mu = 0$, the above equation becomes the Gauss-Newton method. The μ parameter is automatically adjusted at each iteration to secure convergence.

The Levenberg-Marquardt algorithm can be shown according to the following steps:

1. Initialize the weights and the learning parameters to small numbers for the first iteration ($t = 1$).

2. Compute the sum of the squared errors over all training patterns as in equation (2.1.5).

3. Use equation (2.1.10) to obtain the increment of the weights Δw.

4. Use equation (2.1.9) to obtain the new weights.

5. Recompute the cost function (equation (2.1.5)) with the new weights.

6. If the current total error has increased, then increment the step to $(t+1)$ (such as reset the weight vector to the previous value) and increase the learning parameter μ, and perform the update again.

7. If the current total error is decreased, then accept the step and decrease the learning parameter μ.

8. Go to step 5 with the new weights until the current total error is smaller than the required value.

Calculation of the Jacobian Matrix (J)

The Jacobian matrix in equation (2.1.11) calculates the first-order partial derivatives of the error with respect to the weights. Forward calculation from the input layer to the hidden layer, and from the hidden layer to the output layer can be now described below.

Consider the weight $w_{i,j}$ of the input layer i to the hidden layer j, $w_{j,k}$ of the hidden layer j to the output layer k. Also, let out_j denote the output of the hidden layer. The hidden layer activation function is selected as a sigmoidal function and denoted by $f(.)$. The NARX network input at time t is the total of the present and the past values of the input and the present and the past values of the estimated output. Therefore, the input to the network is $I_i(t)$, $i = 1, 2, 3, ..., M$ where M is the total input and output delays, and the output at time t is denoted by $y_k(t)$, $k = 1, 2, 3, ..., K$. Assume that the NARX network iterations are denoted by t, so that one calculates [173]:

$$net_j(t) = \sum_{i=1}^{M} w_{i,j} I_i(t) \tag{2.1.12}$$

$$out_j(t) = f(net_j(t)) \tag{2.1.13}$$

$$net_k(t) = \sum_{j=1}^{H} w_{j,k} out_j(t) \tag{2.1.14}$$

where H denotes the number of hidden layer neurons. To minimize the cost function F, one needs to compute:

$$\Delta w \propto -\frac{\partial e}{\partial w} \tag{2.1.15}$$

At iteration t for the output layer with a particular weight calculated from the forward computation we have

$$\Delta w_{k,j}(t) \propto -\frac{\partial e_k(t)}{\partial w_{k,j}(t)} \tag{2.1.16}$$

However, the error is not directly a function of the weight. Therefore, by using the chain rule and expanding equation (2.1.16), we have:

$$\Delta w_{k,j}(t) = -\frac{\partial e_k(t)}{\partial y_k(t)} \frac{\partial y_k(t)}{\partial net_k(t)} \frac{\partial net_k(t)}{\partial w_{k,j}(t)} \tag{2.1.17}$$

$$\frac{\partial e_k(t)}{\partial y_k(t)} = \frac{\partial 1/2(d_k(t) - y_k(t))^2}{\partial y_k(t)} = -(d_k(t) - y_k(t)) \tag{2.1.18}$$

$$\frac{\partial y_k(t)}{\partial net_k(t)} = \frac{\partial (1 + exp^{-net_k(t)})^{-1}}{\partial net_k} = \frac{exp^{-net_k(t)}}{(1 + exp^{-net_k(t)})^2} = y_k(t)(1 - y_k(t)) \tag{2.1.19}$$

$$\frac{\partial net_k(t)}{\partial w_{k,j}(t)} = \frac{\partial(w_{k,j}(t)out_j(t))}{\partial w_{k,j}(t)} = out_j(t) \tag{2.1.20}$$

The weight update rule for a hidden to the output weight:

Based on the above derivation the weights are now adjusted according to the following,

$$\Delta w_{k,j}(t) = (d_k(t) - y_k(t))y_k(t)(1 - y_k(t))out_j \tag{2.1.21}$$

Equation (2.1.21) can be simplified as shown in equation (2.1.22), where the δ term represents the product of the error with the derivative of the activation function as

$$\Delta w_{k,j}(t) = \delta_k(t)out_j(t) \tag{2.1.22}$$

with

$$\delta_k(t) = (d_k(t) - y_k(t))y_k(t)(1 - y_k(t)) \tag{2.1.23}$$

The weight update rule for an input to the hidden weight:

Similarly, the weights are adjusted as follows,

$$\begin{aligned}\Delta w_{j,i}(t) = -[\sum_k \frac{\partial e_k(t)}{\partial y_k(t)} \frac{\partial y_k(t)}{\partial net_k(t)} \frac{\partial net_k(t)}{\partial out_j(t)}] \frac{\partial out_j(t)}{\partial net_j(t)} \frac{\partial net_j(t)}{\partial w_{i,j}(t)} = \\ [\sum_t \delta_k w_{k,j}(t)] out_j(t)(1 - out_j(t))u_i = \delta_j I_i\end{aligned} \tag{2.1.24}$$

Based on the information given about the NARX structure, the neural network's input includes feedback from the network output. Therefore, this network has the capability to capture the dynamics of a non-linear dynamical system and it can represent the system dynamic features. This network has a simple architecture. The connections between neurons are indicated with the weights. These weights are updated and the errors are calculated. Iterations are preformed until the desired error

for the network is obtained. The proposed approach will be evaluated in Chapter 3 for multi-steps ahead prediction. The jet engine turbine temperature will be predicted using different simulations in this thesis.

2.2 Elman Network

An Elman network [163] is in principle a regular feed-forward network with local feedback which is used to build memory in the system. It consists of three layers of input, hidden and output layers. In contrast to the NARX network, the output is not fed back to the input layer. Otherwise, special units called context units save previous output values of the hidden layer neurons. These units are hidden in the sense that they interact exclusively with other nodes internal to the network, and not the outside world. These values are then fed back to the input layer as an additional input to the system [111].

At time (t), the input units receive the first input in the sequence which might be a single scalar value or a vector depending on the nature of the problem. Both the input units and context units activate the neurons in the hidden layer. The hidden neurons then activate the output neurons. They are also fed back to activate context units. The output is compared with the actual ones and back propagation of error is used to adjust the weights. At the next time step $(t+1)$, this sequence is repeated. At this time, the context units contain units which are exactly the hidden neuron values at time (t). These context units thus provide the network with memory [163]. Elman network with three layers is shown in Figure 2.3.

In this network, the neurons of the input layer, hidden layer and output layer are fully connected by weight matrices. Context units which save the previous values of the hidden neurons are also connected to the hidden layer through connection weights. Based on this methodology, the network output is related to the current input data

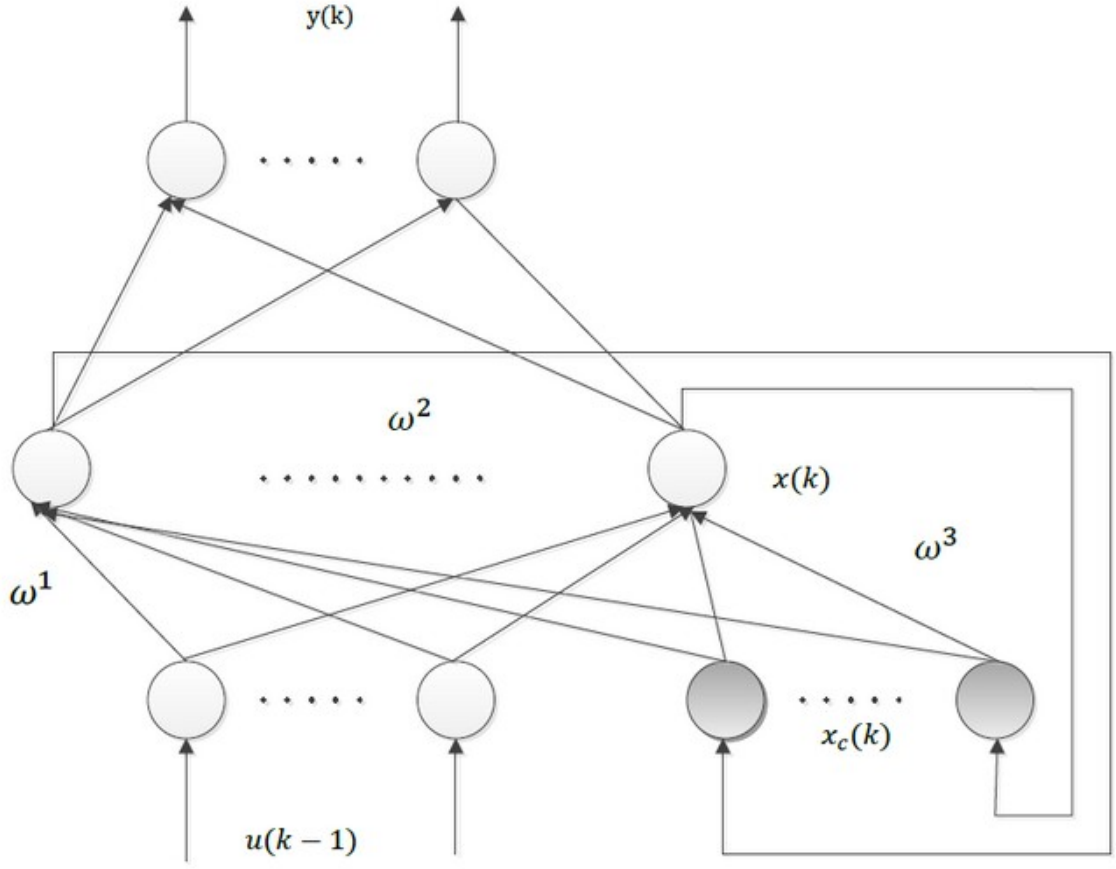

Figure 2.3: Elman network architecture [112].

as well as historical input data due to the context neurons. This implies that the output is a function of both previous activation stated and current data [117].

2.2.1 Elman Network Learning Algorithm

As mentioned in Section 2.2, current inputs of the hidden layer consist of input signals which go to the hidden layer through connection weights w and previous time steps of the hidden layer fed back to the input layer. After the signal is processed in the hidden layer, it is sent to the output layer where a decision is made as to whether the output is expected or not. If the output differs from the expected one, the error returns along the original connection path. This iteration is performed until the desired error for the network is achieved. Elman network error at any time (t) is evaluated according to the following error squared function [112]:

$$e(t) = 1/2 \sum_{k=1}^{K} (d_k(t) - y_k(t))^2 \tag{2.2.1}$$

where $d_k(t)$ is the desired value of the k^{th} output neuron and $y_k(t)$ is the output of the Elman network at time t, and K is the number of the output layer neurons.

Consider the weight $w_{i,j}$ of the input layer i to the hidden layer j, $w_{j,k}$ of the hidden layer j to the output layer k, and weight $w_{l,j}$ of context unit l to the hidden layer j. The context layer neuron input defines as $netC_l$. Also, out_j is the output of the hidden layer. Hidden layer activation function $f(.)$ is a sigmoidal function. Elman network input and output are recorded as $u_i(t)$, $i = 1, 2, 3, ..., N$ and $y_k(t)$, $k = 1, 2, 3, ..., K$ respectively, where $u_i(t)$ and $y_k(t)$ are sequential input and output data. Assume the Elman network iterations are denoted by $t, t = 1, 2, 3, ..., T$ so that one calculates:

$$net_j(t) = \sum_{i=1}^{N} w_{i,j} u_i(t) + \sum_{l=1}^{L} w_{l,j} netC_l(t) \tag{2.2.2}$$

where L is the number of context layer nodes which is equal to the hidden layer nodes.

$$out_j(t) = f(net_j(t)) \tag{2.2.3}$$

$$net_k(t) = \sum_{j=1}^{L} w_{j,k} out_j(t) \tag{2.2.4}$$

where the t^{th} context layer neuron input $netC_l(t)$ is the same as the $(t-1)^{th}$ neuron output of the hidden layer of $out_j(t-1)$, that is

$$netC_l(t) = out_j(t-1) \tag{2.2.5}$$

This network still uses the back propagation principle of the gradient decent error feedback, where the difference is the indicator function that is defined as the overall approximation error within a time interval $[0, T]$, as follows

$$e = 1/2 \sum_{t=1}^{T} \sum_{k=1}^{K} (d_k(t) - y_k(t))^2 \tag{2.2.6}$$

where $d_k(t)$ is the desired output of the network at time t and $y_k(t)$ is the network output at time t. To minimize the function e, one needs to compute:

$$\Delta w \propto -\frac{\partial e}{\partial w} \tag{2.2.7}$$

At time t for the output layer with a particular weight calculated from the forward computation there is

$$\Delta w_{k,j}(t) \propto -\frac{\partial e}{\partial w_{k,j}(t)} \tag{2.2.8}$$

However, the error is not directly a function of the weight. Therefore, by using the chain rule and expanding equation (2.2.8), we have:

$$\Delta w_{k,j}(t) = -\frac{\partial e(t)}{\partial y_k(t)} \frac{\partial y_k(t)}{\partial net_k(t)} \frac{\partial net_k(t)}{\partial w_{k,j}(t)} \tag{2.2.9}$$

$$\frac{\partial e(t)}{\partial y_k(t)} = \frac{\partial 1/2(d_k(t) - y_k(t))^2}{\partial y_k(t)} = -(d_k(t) - y_k(t)) \tag{2.2.10}$$

$$\frac{\partial y_k(t)}{\partial net_k(t)} = \frac{\partial (1 + exp^{-net_k(t)})^{-1}}{\partial net_k} = \frac{exp^{-net_k(t)}}{(1 + exp^{-net_k(t)})^2} = y_k(t)(1 - y_k(t)) \tag{2.2.11}$$

$$\frac{\partial net_k(t)}{\partial w_{k,j}(t)} = \frac{\partial (w_{k,j}(t) out_j(t))}{\partial w_{k,j}(t)} = out_j(t) \tag{2.2.12}$$

The weight update rule for a hidden to the output weight:

Based on the above discussion the weights are now adjusted according the following,

$$\Delta w_{k,j}(t) = (d_k(t) - y_k(t)) y_k(t)(1 - y_k(t)) out_j(t) \tag{2.2.13}$$

Equation (2.2.13) can be simplified as shown in equation (2.2.14), where the δ term represents the product of the error with the derivative of the activation function as

$$\Delta w_{k,j}(t) = \delta_k(t) out_j(t) \tag{2.2.14}$$

with

$$\delta_k(t) = (d_k(t) - y_k(t)) y_k(t)(1 - y_k(t)) \tag{2.2.15}$$

The weight update rule for an input to the hidden weight:

Similarly, the weights are adjusted as follows,

$$\Delta w_{j,i}(t) = -[\sum_{k} \frac{\partial e(t)}{\partial y_k(t)} \frac{\partial y_k(t)}{\partial net_k(t)} \frac{\partial net_k(t)}{\partial out_j(t)}] \frac{\partial out_j(t)}{\partial net_j(t)} \frac{\partial net_j(t)}{\partial w_{i,j}(t)} = \\ [\sum_{t} \delta_k w_{k,j}(t)] out_j(t)(1 - out_j(t)) u_i = \delta_j(t) u_i(t) \tag{2.2.16}$$

The weight update rule for the context unit to the hidden weight:

Similarly, the weights are adjusted as follows,

$$\Delta w_{j,i}(t) = -[\sum_{k} \frac{\partial e(t)}{\partial y_k(t)} \frac{\partial y_k(t)}{\partial net_k(t)} \frac{\partial net_k(t)}{\partial out_j(t)}] \frac{\partial out_j(t)}{\partial net_j(t)} \frac{\partial net_j(t)}{\partial w_{i,j}(t)} = \\ [\sum_{k} \delta_k(t) w_{k,j}(t)] out_j(t)(1 - out_j(t)) netC_l(t) = \delta_j(t) netC_l(t) \tag{2.2.17}$$

Based on the information given about the Elman neural network architecture, the neurons in the context layer hold a copy of the output of the hidden neurons. The value of these neurons are used as an extra input for all the hidden neurons in time steps later. Therefore, this feedback is used to construct memory in the network. For this reason, this network is suitable for learning the dynamics of non-linear systems. It is trained with gradient descent backpropagation. The weight values are optimized during the stage of training until the difference between the actual output and the network's output becomes lower than the desired one. The performance of this network will be tested in jet engine's turbine temperature prediction by using several scenarios in Chapter 4.

2.3 Performance Evaluation

There are many ways to evaluate the performance of a model representation. Mean squared error (MSE) is one of the ways to quantify the difference between values estimated and actual values of the quantity being estimated. This measure of the average of the squared of errors used in [104] and [87]. Zhang *et al.* [114] used normalized mean squared error (NMSE) to calculate the prediction error of the neuro-fuzzy system. MSE is used in [115] along with the mean absolute error (MAE) and mean absolute percentage error (MAPE). MSE and mean absolute deviation (MAD) are selected to evaluate the forecasting method by Zhang [67]. Root mean squared error (RME) is another way of evaluating the representation performance which is used extensively in the literature [116, 117, 107] . Shen *et al.* [94] evaluated the performance of a multi-step prediction with NARX network using RMSE [94]. RMSE is also employed to evaluate forecasting capability in [118].

In this thesis, standard deviation of error, mean of the error and RMSE are used to evaluate the performance of the network. In order to calculate the RSME, the difference between estimated and actual values are each squared and then averaged over the sample period and then rooted. The mathematical formula is shown below:

$$RMSE = \sqrt{\frac{\sum_{k=1}^{K}(y_k - \hat{y}_k)^2}{K}}, \tag{2.3.1}$$

where K represents the number of data points to be predicted, y_k is the actual value and $\hat{y}_k$ is the predicted value. The $RSME$ can range from 0 to ∞. It should be noted that lower values show better performance in prediction.

2.4 Uncertainty Management and Prediction Intervals

In gas turbine performance analysis, gas path measurements such as turbine and compressor temperature or pressure are used to recognize that the engine goes through degradations or faults. This fact necessitate the importance of measurements for reliable prognosis. To overcome uncertainty in measurements prediction bounds are introduced [119].

In statistical measures, prediction bounds are the estimate of the upper and the lower bounds which the future observations will fall within these bounds as an ideal way of quantifying the degree of uncertainty around a specific parameter [120].

Lower and upper bounds (denoted by l and u) for a future observation X in a normal distribution with known mean μ and standard deviation σ can be calculated as [121]:

$$\gamma = P(l < X < u) = P(\frac{l-\mu}{\sigma} < \frac{X-\mu}{\sigma} < \frac{u-\mu}{\sigma}) = P(\frac{l-\mu}{\sigma} < Z < \frac{u-\mu}{\sigma}) \quad (2.4.1)$$

where $Z = \frac{x-\mu}{\sigma}$. Hence,

$$\frac{l-\mu}{\sigma} = -z, \frac{u-\mu}{\sigma} = z \quad (2.4.2)$$

or

$$l = \mu - z\sigma, u = \mu + z\sigma \quad (2.4.3)$$

where z is defined as:

$$\gamma = P(-z < Z < z) \tag{2.4.4}$$

or equivalently,

$$P(Z > z) = 1/2(1 - \gamma) \tag{2.4.5}$$

Therefore, the prediction interval can be written as $[\mu - z\sigma, \mu + z\sigma]$.

There are different values associated with z for different probabilities. These values are shown in Table 2.1.

Table 2.1: z value for different probabilities [122].

Probability Percentage	z
50%	0.67
90%	1.64
95%	1.96
99%	2.58

2.5 Jet Engine Mathematical Model

Gas turbine engines are used in many industrial and aerospace applications. One kind of gas turbine called jet engine is a reaction engine used to generate high-speed thrust by jet propulsion in accordance with Newton′s laws of motion.

Based on the work of Naderi *et al.* [123] on the modelling of an aircraft jet engine, a MATLAB/Simulink model for a single spool jet engine is developed. The simulation model was developed by using mechanical, aerodynamic and thermodynamic relationships between the components of the system. The information flow among different parts of a single spool jet engine is shown in Figure 2.4.

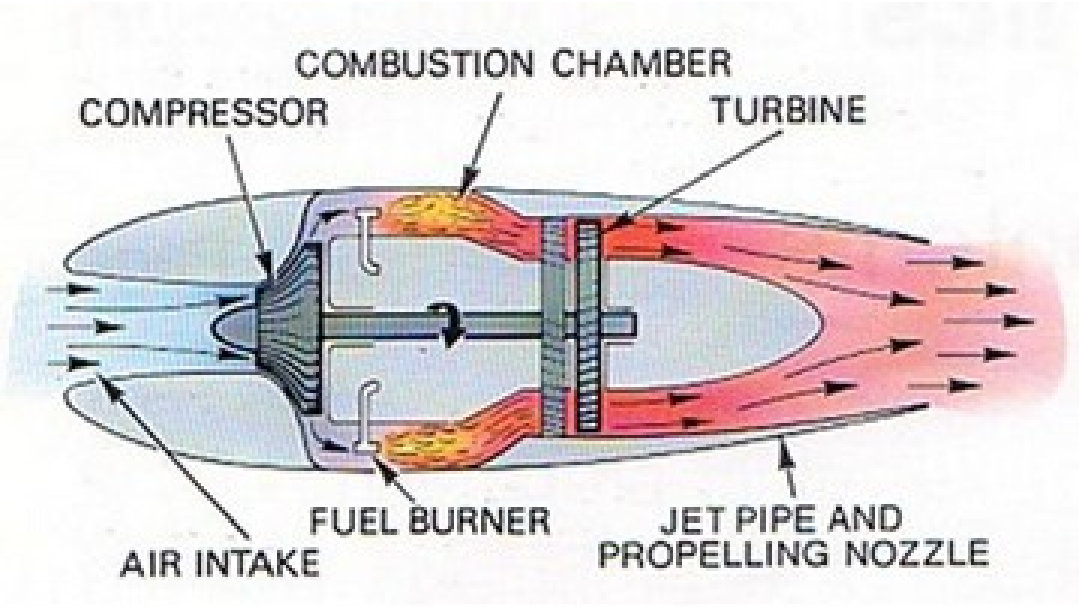

Figure 2.4: Schematic view of a single-spool jet engine [124].

Different components of a single spool jet engine can be described as:

- Air intake (Diffuser)

  In the jet engine, a proportion of the incoming air goes through the air intake and diffuser to decelerate the air relative to the engine where a temperature and pressure rise known as a ram effect is associated with this deceleration. By assuming adiabatic process (process where there is no heat exchange between the system and the environment) the temperature and pressure relative to the environment can be computed as:

$$\frac{P_d}{P_{amb}} = [1 + \eta_d \frac{\gamma - 1}{2} M^2]^{\frac{\gamma}{\gamma - 1}}, \tag{2.5.1}$$

$$\frac{T_d}{T_{amb}} = 1 + \frac{\gamma - 1}{2} M^2, \tag{2.5.2}$$

where P_d denotes diffuser pressure, P_{amb} is the ambient pressure, η_d denotes the diffuser efficiency, γ denotes the heat capacity ratio, M is the Mach number, T_d diffuser temperature, and T_{amb} denotes the ambient temperature.

- Compressor

  The air goes through the compressor to provide high pressure air for the combustion chamber. Given the pressure ratio (π_c) and the corrected rotational speed ($\frac{N}{\sqrt{\theta}}$), one can obtain the corrected mass flow rate ($\frac{\dot{m}_C\sqrt{\theta}}{\delta}$) and efficiency ($\eta_C$) from the performance map by using a proper interpolation technique, where ($\theta = \frac{T_i}{T_o}$) and ($\delta = \frac{P_i}{P_o}$), i.e. ($\frac{\dot{m}_C\sqrt{\theta}}{\delta} = f\dot{m}_C(\frac{N}{\sqrt{\theta}}, \pi_C)$) and ($\eta_C = f\eta_C(\frac{N}{\sqrt{\theta}}, \pi_C)$).

  When these parameters are obtained, the compressor temperature rise and the mechanical power are calculated as follows [125]:

$$\frac{T_o}{T_i} = [1 + \frac{1}{\eta_C}(\pi_C{}^{\frac{\gamma-1}{\gamma}} - 1)], \tag{2.5.3}$$

  where T_o is the compressor output temperature and T_i is the compressor input temperature.

$$W_C = \dot{m}_C C_p (T_o - T_i), \tag{2.5.4}$$

  It should be noted that the power consumed by the compressor (W_C) is related to the speed of the shaft $w_C = \frac{J\frac{N\cdot 2\pi}{60}^2}{2}$ where J is the momentum of inertia of the shaft and N is the speed of the shaft. C_p is the specific heat at constant pressure, and $\dot{m}_C$ is the compressor mass flow rate.

- Combustion Chamber

  The combustion chamber represents both the energy accumulation and volume dynamics between compressor and turbine. Fuel and high pressure air is burned and the temperature is increased. Combustion chamber dynamics can be represented as

$$\dot{P}_{CC} = \frac{P_{CC}}{T_{CC}}\dot{T}_{CC} + \frac{\gamma R T_{CC}}{V_{CC}}(\dot{m}_C + \dot{m}_f - \dot{m}_T), \tag{2.5.5}$$

$$\dot{T}_{CC} = \frac{1}{c_\nu m_{CC}}[(c_p T_C \dot{m}_C + \eta_{CC} H_u \dot{m}_f - c_p T_{CC} \dot{m}_T) - c_\nu T_{CC}(\dot{m}_C + \dot{m}_f - \dot{m}_T)], \tag{2.5.6}$$

  where P_{CC} is the combustion chamber pressure, T_{CC} denotes the temperature in the combustion chamber, and V_{CC} the volume of the combustion chamber. R denotes the Reynold′s number, $\dot{m}_C$ compressor mass flow rate, $\dot{m}_f$ fuel mass flow rate, $\dot{m}_T$ turbine mass flow rate, c_ν specific heat at constant volume, η_{CC} the combustion chamber efficiency, and H_u is the fuel specific heat.

- Turbine

  Kinetic energy and high temperature released in the combustion chamber due to the air and fuel burning is now used to drive the compressor and accessories which cause a drop in the temperature. However, it should be noted that a high portion of produced power is used internally to drive the compressor. Saravanamuttoo *et al.* [126] concluded that around 75% of the power is required to drive the compressor. Based on [125], having pressure ratio (π_T) and the corrected rotational speed ($\frac{N}{\sqrt{\theta}}$), the corrected mass flow rate ($\frac{\dot{m}_T\sqrt{\theta}}{\delta}$) and the efficiency ($\eta_T$) are calculated from the performance maps (these maps are from

the software package (GSP) [164]), i.e. $(\frac{\dot{m}_T\sqrt{\theta}}{\delta} = f\dot{m}_T(\frac{N}{\sqrt{\theta}}, \pi_T))$.

The temperature drop and the mechanical power are

$$\frac{T_o}{T_i} = [1 - \eta_T(1 - \pi_T)^{\frac{\gamma-1}{\gamma}}], \tag{2.5.7}$$

$$W_T = \dot{m}_T C_p(T_i - T_o), \tag{2.5.8}$$

where T_o is the turbine output temperature, T_i denotes the temperature in the turbine inlet, and W_T is the power generated by turbine.

- Nozzle

  The gas leaves the turbine at a pressure greater than atmosphere. Thus, it goes through a nozzle to decrease its pressure. It also expands to higher velocity before being discharged to the environment to produce thrust. The nozzle exit temperature T_{n0} is given by

$$T_{ni} - T_{no} = \eta_n T_{no}[1 - (\frac{1}{\frac{P_{ni}}{P_{amb}}})^{\frac{\gamma-1}{\gamma}}], \tag{2.5.9}$$

  where T_{ni} denotes the nozzle inlet temperature, and T_{no} denotes the nozzle output temperature, η_n is the nozzle efficiency, P_{ni} denotes the nozzle inlet pressure, and P_{amb} is the pressure of the ambient.

  Rotor and volume dynamics are considered to obtain the non-linear dynamics of the system. The engine components are considered to model an imbalance mass flow rate for developing volume dynamics of the system. Heat transfer dynamics is caused by considerable differences between the air stream temperature and components temperature due to a large power excursion, e.g., during start up

or rapid maneuvers of an aircraft [127]. This effect is not considered since this model is concerned with commercial single spool jet engine at normal operating conditions. A schematic depicting the main modules and the overall information flows are shown in Figure 2.5.

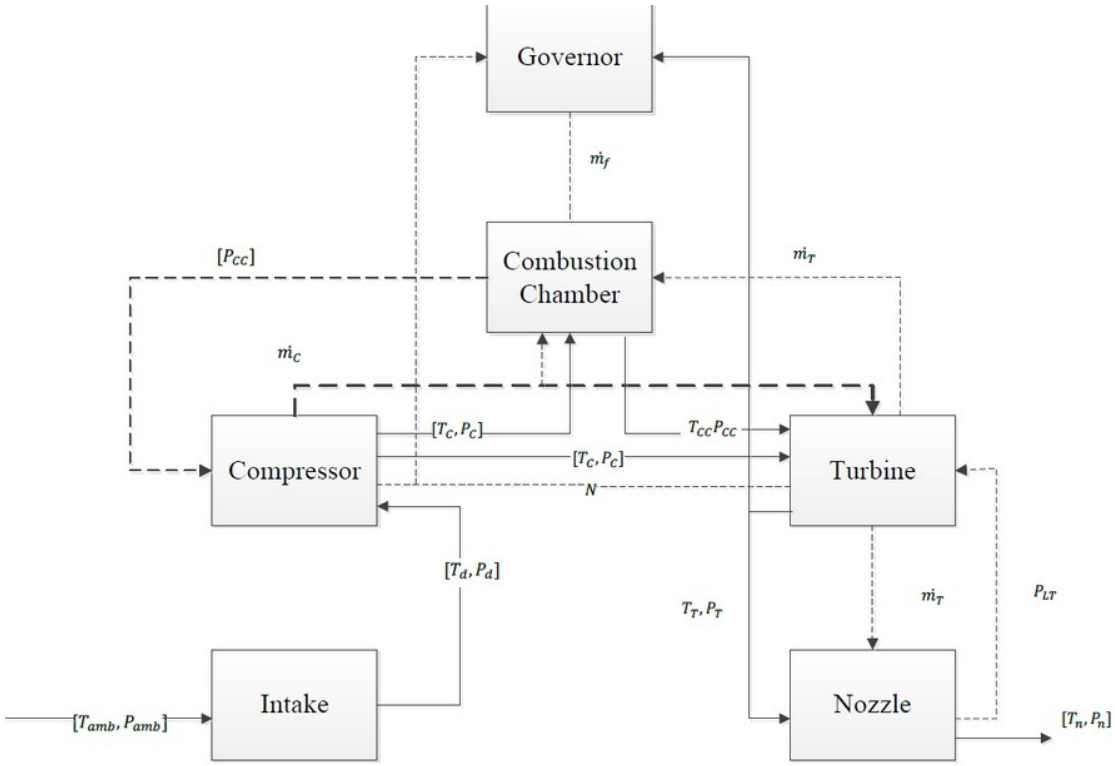

Figure 2.5: The aircraft jet engine modules and information flow chart [123].

- Rotor Dynamics

  Energy balance between the shaft and the compressor contributes to the following equation where E is the energy of the turbine, η_{mech} denotes the mechanical efficiency, and power of the turbine and the compressor represented by W_T and W_C respectively.

$$\frac{dE}{dt} = \eta_{mech} W_T - W_C \tag{2.5.10}$$

- Volume Dynamics

  The unbalance mass flow rates between different components are considered by the volume dynamics. Assuming that the gas has zero speed and homogeneous properties over volumes, it can be defined by the following equation:

$$\dot{P}V = RT(\sum \dot{m}_{in} - \sum \dot{m}_{out}), \tag{2.5.11}$$

  where V is the volume, P is the pressure, R denotes the Reynold′s number, T temperature, $\dot{m}_{in}$ denotes the inlet mass flow rate, and $\dot{m}_{out}$ denotes the outlet mass flow rate.

Now, the temperature, the pressure, and the rotational speed can be obtained from the above non-linear equations for each component. The set of non-linear equations corresponding to a single spool jet engine is obtained by [123] which show that our system is a non-linear system of 4^{th} order given by

$$\dot{T}_{CC} = \frac{1}{c_\nu m_{CC}}[(c_P T_C \dot{m}_C + \eta_{CC} H_u \dot{m}_f - c_P T_{CC} \dot{m}_T) - c_\nu T_{CC}(\dot{m}_C + \dot{m}_f - \dot{m}_T)], \tag{2.5.12}$$

$$\dot{N} = \frac{\eta_{mech} \dot{m}_T c_P (T_{CC} - T_T) - \dot{m}_C c_P (T_C - T_d)}{JN(\frac{\pi}{30}^2)}, \tag{2.5.13}$$

$$\dot{P}_T = \frac{RT_{Mi}}{V_{Mi}}(\dot{m}_T + \frac{\beta}{1+\beta}\dot{m}_C - \dot{m}_n), \tag{2.5.14}$$

$$\dot{P}_{CC} = \frac{P_{CC}}{T_{CC}}\dot{T}_{CC} + \frac{\gamma R T_{CC}}{V_{CC}}(\dot{m}_C + \dot{m}_f - \dot{m}_T), \tag{2.5.15}$$

where T_{CC} is the temperature in the combustion chamber, N defines the rotor speed, m_{CC} is the mass flow in the combustion chamber, c_ν is the specific heat at constant volume, c_p denotes the specific heat at constant pressure, T_C denotes the compressor temperature, $\dot{m}_C$ denotes the compressor mass flow rate, η_{CC} is the combustion chamber efficiency, H_u denotes fuel specific heat, $\dot{m}_f$ denotes the fuel flow mass flow

rate, η_{mech} denotes the mechanical efficiency, T_d denotes the diffuser temperature, $\dot{m}_n$ denotes the mass flow rate in the nozzle, β is the bypass ratio, T_{Mi} is the mixer temperature, and V_{Mi} is the volume of the mixer, and P_{CC} denotes the combustion chamber pressure.

The input of the single spool engine is the power level angle (PLA) which is related to the mass flow rate through a variable gain. Dynamics for the fuel mass flow rate is shown as

$$\tau \frac{d\dot{m}_f}{dt} + \dot{m}_f = G u_{fd}, \tag{2.5.16}$$

where τ denotes the time constant, G is the gain associated with fuel valve, and u_{fd} denotes the fuel demand which is computed by using a feedback from the rotational speed [165].

The engine model in this thesis has seven (7) measurements namely compressor temperature, compressor pressure, combustion chamber temperature, combustion chamber pressure, rotor speed, turbine pressure, and finally turbine temperature. However, practically and being as close as possible to a realistic engine, it is hard to measure the temperature of combustion chamber due to high temperature and chemical activities inside the combustion chamber [128], so the engine measurable parameters decrease to six (6).

The engine goes through different operating regimes in each flight, namely starting thrust, take-off, acceleration, climbing, cruise, deceleration, shutdown, etc. A simple aircraft mission profile is depicted in Figure 2.6.

2.5.1 Engine Data Generation

The data used in this thesis are generated by using the MATLAB/Simulink model based on the equations described in the previous section as well as those in [123]. The

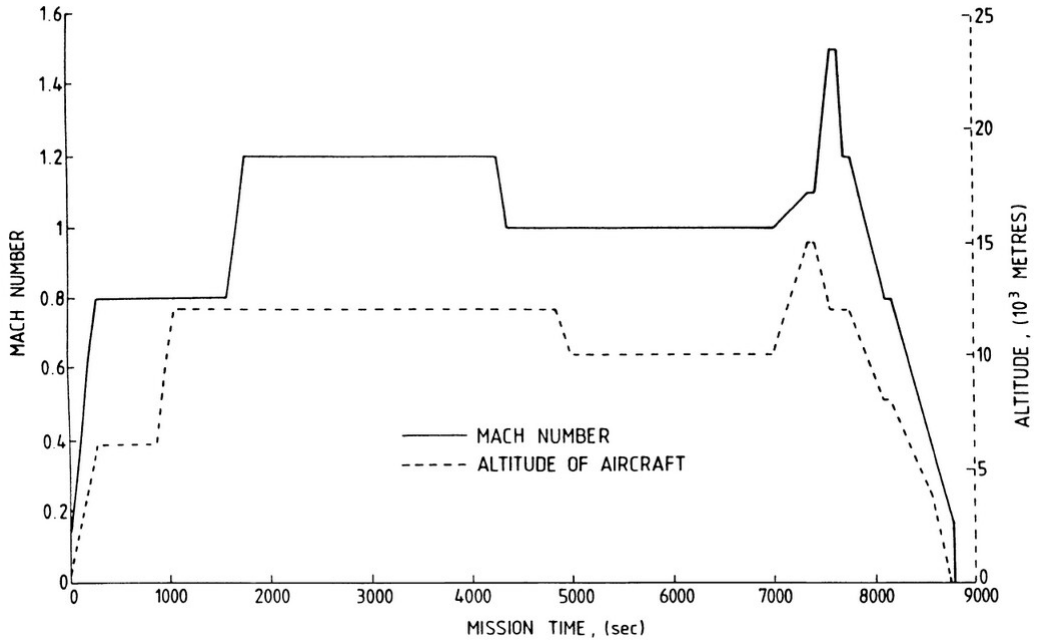

Figure 2.6: Aircraft mission profile [1].

engine model is operating in its take-off mode when the fuel usage is in its maximum value and maximum thrust is provided. The Mach number changes from $0 - 0.2$ in 20 seconds of engine simulation time. The fuel changes in order to maintain the thrust generated by the engine constant. The thrust is measured based on the egines pressure ratio (EPR) which is the ratio between exhaust pressure and pressure entering the compressor [1]. The initial ambient parameters are set corresponding to the take-off mode. This leads to the ambient condition of $T_{amb} = 288K$ and $P_{amb} = 1.0133$ atmosphere. In order for the model to be as close as possible to the practical engine operating measurement, noise was added to the system based on Table 2.2.

Table 2.2: Noise standard deviations [107].

C_T	C_P	T_T	T_P	N	$\dot{W}_f$
0.23	0.164	0.097	0.164	0.051	0.51

where C_T denotes the compressor temperature, C_P denotes the compressor pressure, T_T denotes the turbine temperature, T_P denotes the turbine pressure, N is the speed of the rotor, and $\dot{W}_f$ is the fuel flow rate.

2.6 Degradation Modelling

2.6.1 Overview

The function of a gas turbine is the result of the cooperation of different components which effects from tear and wears over time that can affect the operation of the system adversely [131]. Each type of aero-engine deterioration has an adverse effect on the performance of the aircraft resulting in reduced thrust and increased costs [1]. It should be mentioned that due to the variety of operational and design factors for engine component, it is usually difficult to control the speed of degradation [131].

Degradations are usually divided into two main categories; recoverable in which the degradation mechanism can be recovered. These losses can be reversed by operational processes such as keeping the inlet and outlet pressures low, or the losses due to fouling that can be regained by compressor washing. Non-recoverable degradations are the result of mechanical problems which in turn cause damages to the aerofoils. Corrosion, erosion, loss of surface finish on blades, and increased tip clearance are examples of these kinds of deterioration [132]. After these losses occur, the component has to be replaced.

The common degradations in gas turbines can be divided into the following categories

- Fouling

  It is caused by the adherence of particles in the range of $2 - 10\mu m$ or less to aerofoils and surfaces which will in turn contribute to the surface roughness,

changes in aerofoils shape, and it narrows the aerofoil throat aperture [133]. Fouling occurs in both compressor and turbine components of the gas turbine. However, in works such as [134] compressor fouling is considered as the main reason for gas turbine degradation. This degradation will be discussed in more details in Section 2.6.2.

- Corrosion and hot-corrosion

  Corrosion is also another engine component deterioration which is caused by chemical reactions happening among components and contaminants that enter the turbine with the inlet air, fuel injection, water or stream such as salts or reactive gases. This degradation has an adverse effect on the performance of the engine. Corrosion is specially a dominant problem in industrial gas turbines because a lot of industrial gas turbines are located near the sea and sea salt can react with engine components [131]. Corrosion damage of rotating blades is depicted in Figure 2.7.

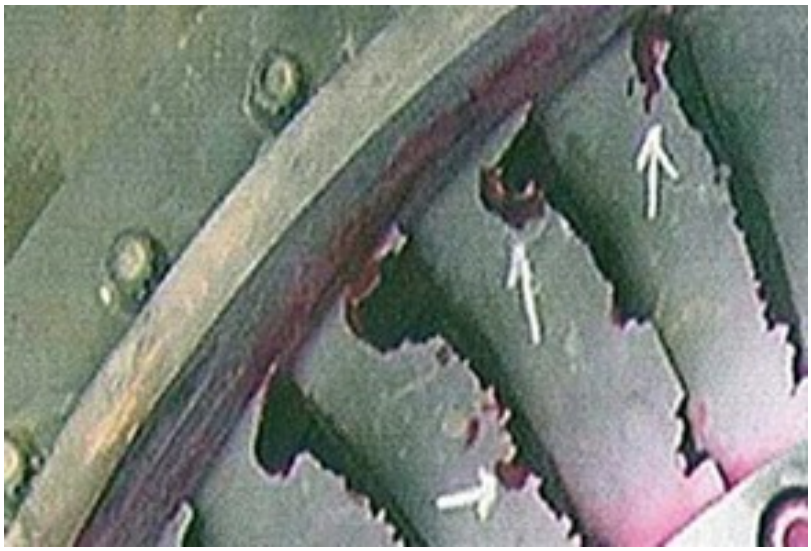

Figure 2.7: Corrosion damage in rotating blades [135].

- High-temperature oxidation

This degradation occurs when metal atoms of components react chemically with oxygen from the environment.

- Erosion

  Erosion is due to the removal of material in the gas flow path causes by particles more than $10\mu m$ in size which leads to changes in aerofoil profiles and throat openings. It can also increase clearances in blades and seals.

- Foreign object damage

  Damage caused by foreign objects is usually because of striking the relatively large objects to the components in the gas path. These objects enter the turbine with the inlet air or gas stream which can contain gravels, bolts, even the birds in aero-engines. On the other hand, pieces of ice produced in the compressor inlet, if are pulled into the path of the gas can cause this problem. This damage could be due to the result of broken off pieces of engine itself [136]. These objects can degrade the engine differently. However, it usually reduces the isentropic efficiency and varies the mass flow rate. It is stated in [137] that the efficiency can reduce to 5% of its real value. This degradation can be recoverable or non-recoverable which necessitate the engine complete shut-down. Figure 2.8 shows an example of the impact of the foreign object to the gas turbine blades.

- Abrasion, rubbing and wearing

  Abrasion is due to the rubbing between rotating and stationary surfaces. This rubbing will increase seal or tip gaps [138].

- Thermal distortion

Figure 2.8: The effect of foreign object damage on gas turbine blades [119].

Hot sections of a gas turbine such as the exit combustion chamber and inlet turbine blades work under high temperature and high-stress fluctuating environment, thus they can distort. These distortions can be seen as twisting, bowing, and welding together of the turbine vanes [138]. High turbine inlet temperature causes damage and distortion in downstream components such as nozzle vanes which in turn increase leakage or creep damage [139]. These distortions can result in permanent failure and increased life-cycle costs [140]. Thermal distortion of blades decreases the isentropic efficiency of the turbine, and variations in the mass flow rate. However, MacLeod *et al.* [139] stated that changes in efficiency are more significant in comparison to mass flow rate.

- Tip clearance

  This deterioration affects both engine's efficiency and flow capacity. It is computed that 0.8% reduction in tip clearance in an engine's compressor reduces the flow capacity by 3% and its efficiency by 2% [141].

Gas turbine efficiency based on the degradation in different parts is depicted in Figure 2.9.

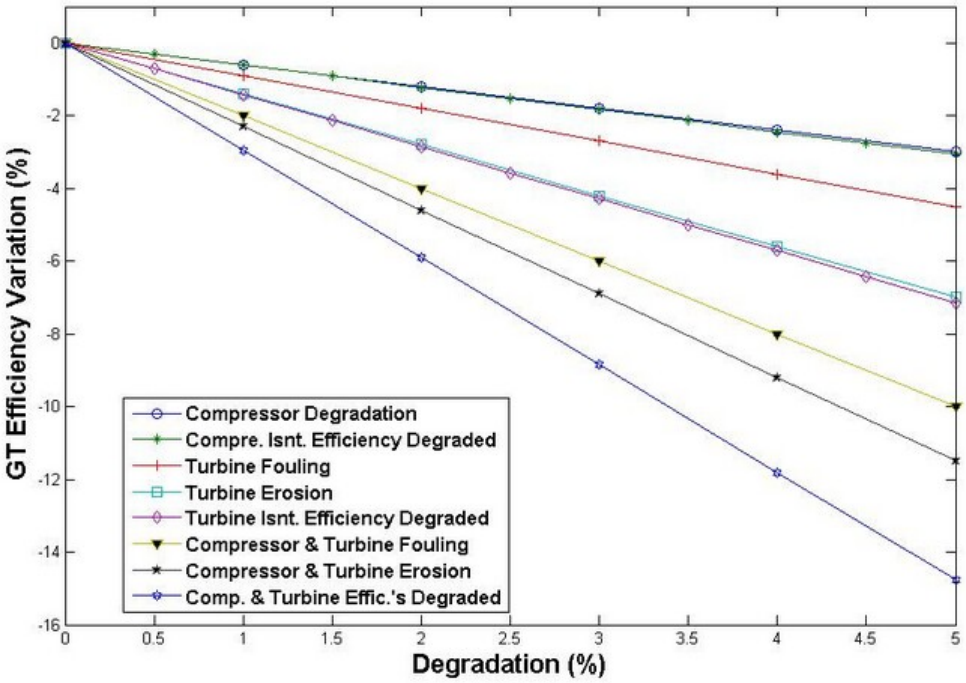

Figure 2.9: Gas turbine efficiency based on component degradations [142].

Degradation occurring in the engine causes a change in the component performance. The degradations introduced by efficiencies and flow capacities in turn can be detected in changes in measurable parameters such as temperature, pressure, speed, etc. With these changes present in the measurements, the problem would be in predicting the future health of the system.

2.6.2 Compressor Fouling

Compressors consume up to 60% of the power produced by turbines, therefore maintaining compressor at its optimum performance during operation is of clear importance. Compressor fouling is one of the main causes of degradation of the performance

of the jet engine that accounts for $70-85\%$ of the total engine performance loss during operations [143]. This degradation can primarily reduce mass flow capacity and compressor's delivery pressure which is then followed by the power reduction and an increase in the heat rate [134]. Mustafa [144] has demonstrated that fouling can reduce mass flow rate by 5% and output power by 13% and an increase in the heat rate by 5.5%. This fact shows the importance of predicting the effects of compressor fouling on the performance of the engine. Fouling is caused by the adherence of particles such as impurities in the air, engine oil leakages or fuel impurities to the compressor blades and consequently, it increases the surface roughness, reduces the flow passage and in some cases changes the shape of the aerofoil [145]. Fouled compressor is shown in Figure 2.10.

Figure 2.10: Fouled compressor [136].

It is shown by Kurz *et al.* [131] that fouling decreases the clearance between the blade and the casing. This reduction causes secondary flows in the section. Compressor fouling can also reduce surge margin which in turn can result in compressor surge [146]. Gulen *et al.* [147] reported that fouling decreases output power by 5% because of mass flow reduction.

All compressors are susceptible to fouling and different factors such as compressor design, aerofoil design and shape, and ambient conditions can affect the rate of fouling [131]. The majority of fouling is caused by particles smaller than $2-10\mu m$. In multi-stage compressors, the effect of fouling is higher in the first stages. According to Wilkinson *et al.* [148] , 70% of fouling in the compressor occurs in its first stages in comparison to rear ones. Bouris *et al.* [149] use numerical study to demonstrate that large particles adhere to the leading edges. The place where the particle adheres differs too. Levine *et al.* [150] reported that fine particles adhere on the rotating blades under high centrifugal forces.

In some cases, fouling is not high enough to damage the gas flow path. In such circumstances, loss can be compensated by on-line/off-line washing and cleaning the surface. Monitoring the fouling deterioration and determining the appropriate time and program for compressor washing will increase safety. It is recommended in [133] that the compressor is washed when the mass flow rate reduces by 2.5%. This solution enables users to use condition-based maintenance instead of periodic strategy to enhance safety factor and decrease costs. However, sometimes fouling has a long-term influence on the performance of the engine, and cannot be removed by only washing [151].

The effects of the compressor fouling are a drop in airflow, pressure ratio, and compressor efficiency which can decrease the power output and thermal efficiency [152]. Fouling decreases the compressor efficiency and mass flow rate. Compressor fouling contributes to a change in the compressor map [151]. A map of a fouled compressor is shown in Figure 2.11.

Various studies predict the influence of compressor fouling on gas turbines performance deterioration. Won Song *et al.* [151] predicted the performance degradation of industrial gas turbine in the presence of compressor fouling. They found that the

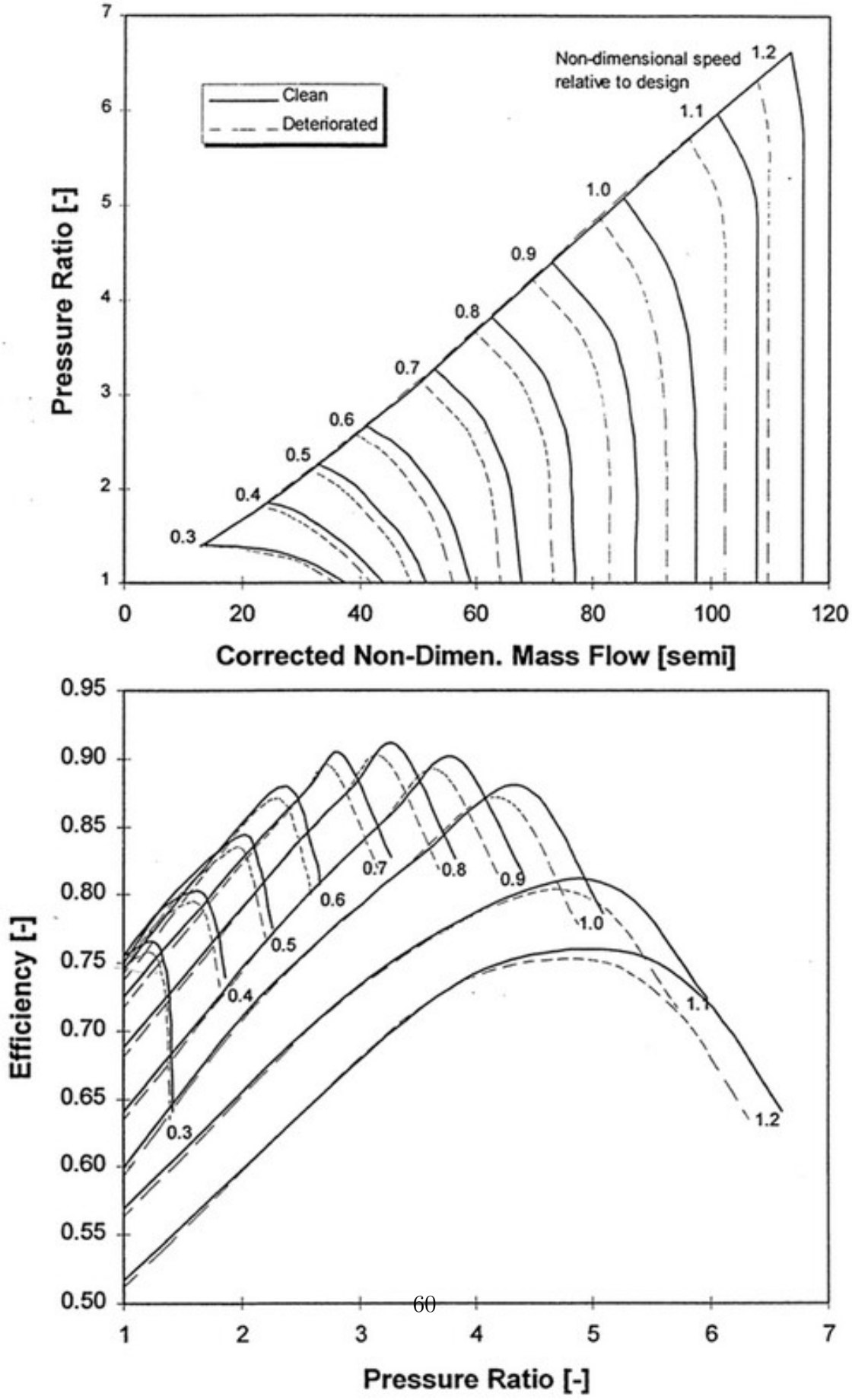

Figure 2.11: Compressor maps in presence of fouling [136].

amount of generated power is strongly sensitive to the fouled conditions of the compressor. Aker *et al.* [134] and Seddigh *et al.* [153] predicted the engine performance by using a linear regression fouling model. Modeling a compressor fouling with exponential behavior was suggested by Tarabrin *et al.* [146]. They modelled the fouling such that it increases exponentially over the time until the thickness of particle deposition stabilizes. Millsaps *et al.* [154] proposed a model to specify the magnitude and location of degradation due to fouling, and the model is checked using a three-stage compressor.

In this thesis, based on the work of Naeem [142], the fouling index (FI) is introduced as a hypothetical parameter to calculate the effects of compressor fouling on its efficiency and mass flow rate. A linear relationship is considered on the health parameters of the system such that 1% fouling in the compressor decreases the efficiency by 1% and causes a reduction of 0.5% in the flow capacity. Rate of change in the compressor mass flow rate ($\Delta\dot{m}_C$), and rate of change in the efficiency of the compressor ($\Delta\eta_C$) for each cycle due to fouling can be calculated as:

$$\Delta\dot{m}_C = 1 - (0.5 * FI/(100 * N)) * i \tag{2.6.1}$$

$$\Delta\eta_C = 1 - (FI/(100 * N)) * i \tag{2.6.2}$$

where i denotes the cycle number and N is the total number of cycles that the compressor fouling will be completed in the specific FI.

Using equations (2.6.1) and (2.6.2) fouling index of 1%, 2% and 3% can be calculated based on Table 2.3 for the total number of fouling cycles equal to 200.

Data is generated from our Simulink model described in Section 2.5 in different fouling indices, namely 1%, 2%, and 3%. The total number of cycles in which the compressor fouling occurs is set to 200. Percentage changes in the engine measurements

Table 2.3: Linear relationship among FI, efficiency and mass flow rate

Fouling Index	$\Delta\eta_C$ reduction	$\Delta\dot{m}_C$ reduction
1%	1%	0.5%
2%	2%	1%
3%	3%	1.5%

are shown in Table 2.4 for different fouling indices.

Table 2.4: Percent change in each measurement for different fouling indices

Fouling Index	$\Delta\%T_T$	$\Delta\%T_P$	$\Delta\%C_T$	$\Delta\%W_f$	$\Delta\%$N
1 %	0.94	0.003	0.5848	0.7182	-0.0142
2 %	1.888	0.003	1.177	1.418	-0.0213
3 %	2.864	0.003	1.784	2.15	-0.0355

where T_T denotes the turbine temperature, T_P denotes the turbine pressure, C_T is the compressor temperature, W_f denotes the fuel flow rate, and N denotes the speed of the shaft.

The results of the compressor fouling effect on measurements are compared together in Figures 2.12-2.15 for $FI = 1\%$, $FI = 2\%$, and $FI = 3\%$. It can be seen that turbine and compressor output temperature increase due to compressor fouling, while rotor speed decreases its value. Fuel flow rate is increased to maintain the output thrust in the constant value. These measurements are more sensitive in higher fouling indices. There is no sudden fracture or failure in the presence of fouling. However, the component may degrade to such an extent that it requires replacement. This fact necessitates the problem of prediction of compressor health measures for health monitoring of jet engines.

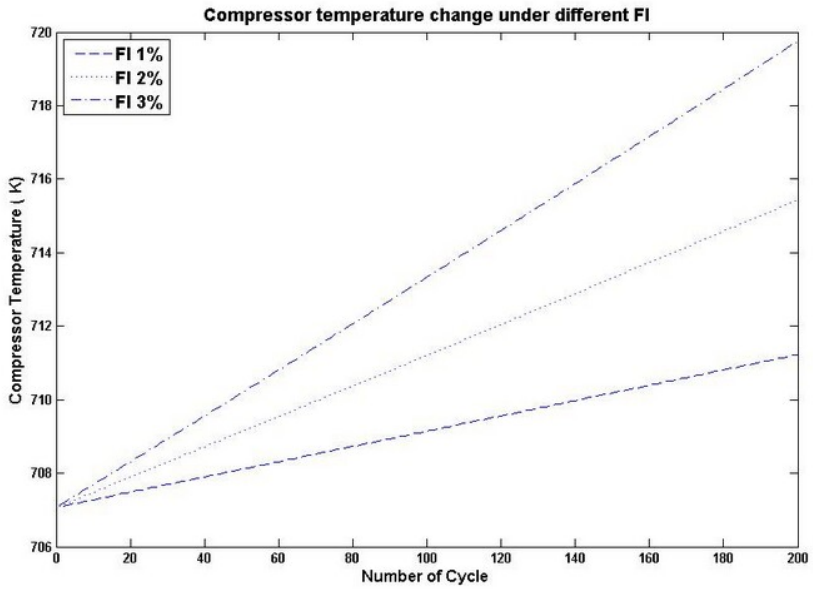

Figure 2.12: Compressor temperature change under different fouling scenarios for the model.

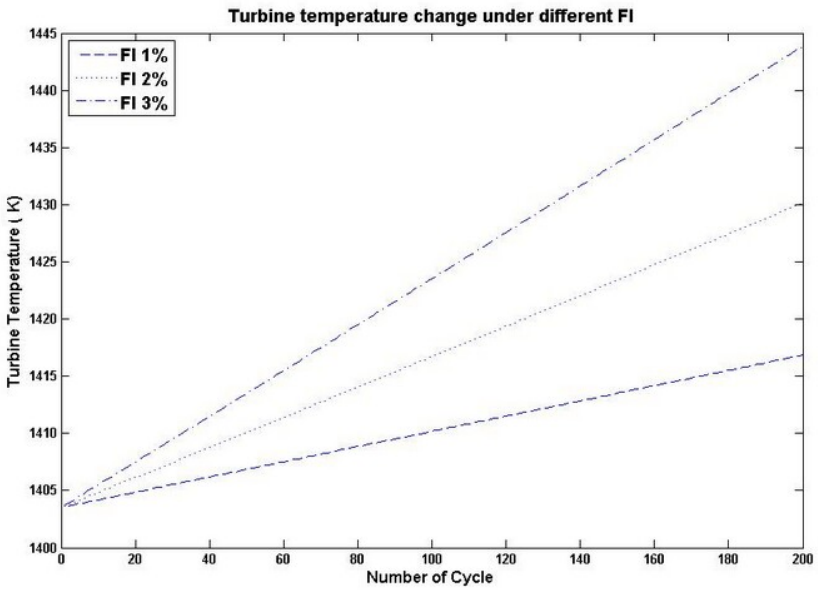

Figure 2.13: Turbine temperature change under different fouling scenarios for the model.

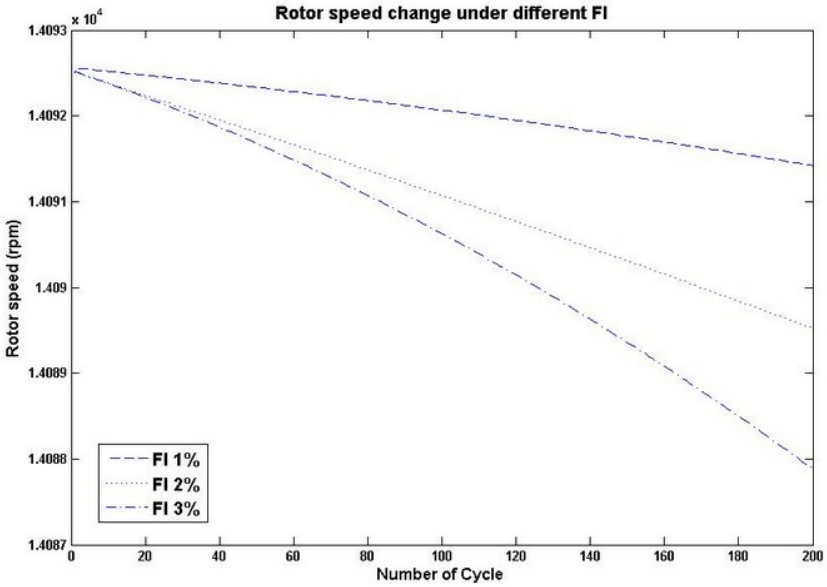

Figure 2.14: Rotor speed change under different fouling scenarios for the model.

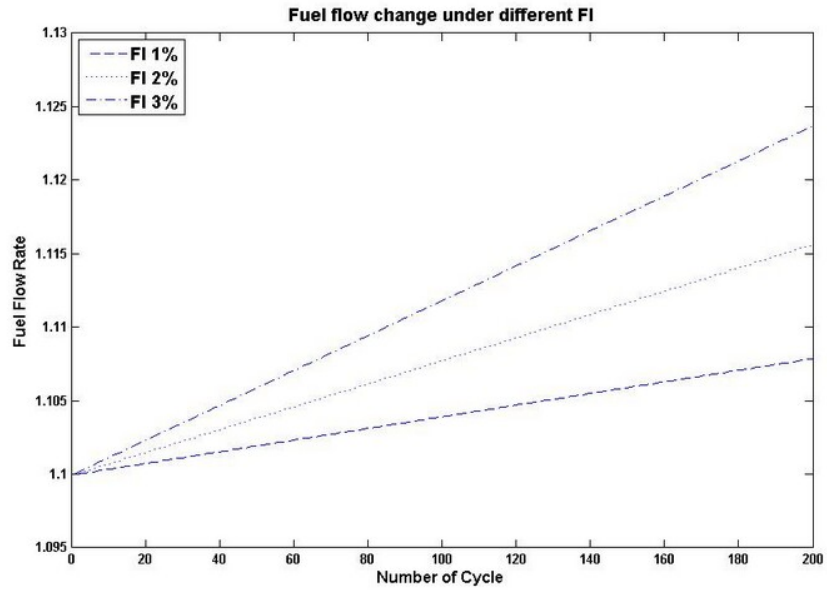

Figure 2.15: Fuel flow rate change under different fouling scenarios for the model.

2.6.3 Combustion Chamber Degradation

Combustion chamber has a low level of degradation in comparison to other sections of a gas turbine. However, if the combustion process varies, it can change the turbine entry temperature which can then affect compressor performance due to high temperature in this section [155]. Variations in the combustor lead to differences in the radial temperature distribution at the entry to the turbine. This can result in localized elevated temperatures, flow-area decreases, greater leakages, increased clearances and distortions. These will reduce efficiency and remaining life of the turbine [140]. It should be noted that variations in combustion chamber temperature does not affect the turbine performance directly. In other words, combustion efficiency does not change with time [136].

2.6.4 Turbine Erosion

Erosion is the loss of material from the flow path by hard particles typically larger than $10\mu m$ which is one of the main causes of deterioration in the turbine section of aero engine applications since aircraft engines are typically exposed to the ingestion of sand or runway materials [138]. Figure 2.16 shows the effect of erosion on turbine blades of a gas turbine engine [136]. Erosion decreases the turbine efficiency and increases the mass flow rate. Erosion is more important in aero engine applications, since the particles larger than $10\mu m$ in diameter is generally eliminated in industrial engines using filtration system [131].

Hamed *et al.* [156] stated that blade erosion has more effect on the compressor adiabatic efficiency in comparison to the pressure ratio [156]. The rate of erosion is highly dependent on the turbine geometry, blade surface material, and particle characteristics [157]. Water drop erosion on turbine blades is modelled numerically in [158]. Metwally *et al.* [159] studied and predicted blade erosion and surface deterioration

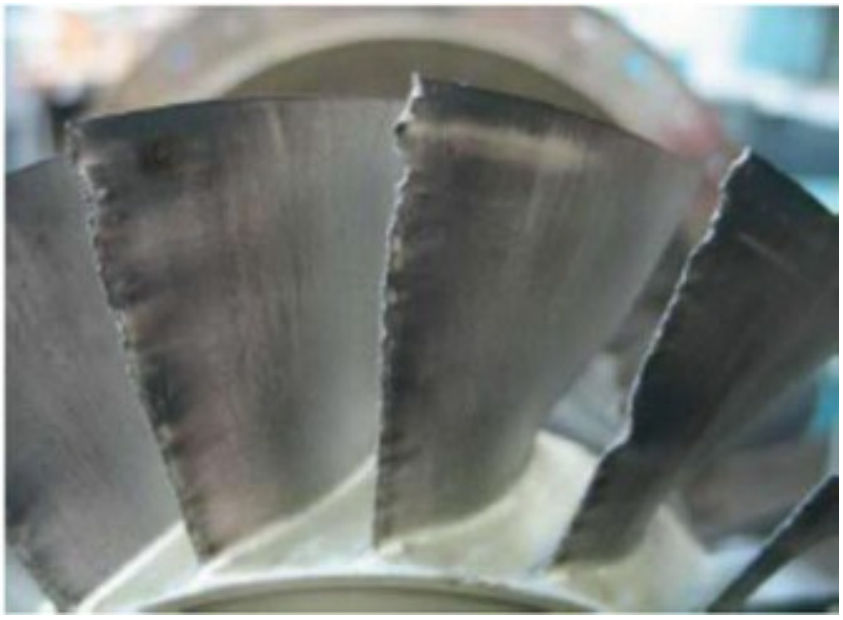

Figure 2.16: The effect of erosion on turbine blades [136].

of the turbine in an automotive gas turbine engine.

A typical turbine map in presence of erosion is shown in Figure 2.17. Based on the work of Naeem [142], it is assumed in this thesis that linear relationship exists between erosion index (EI), turbine efficiency and mass flow rate in which 1% erosion in the turbine decreases the efficiency by 1% and causes an increase of 0.5% in flow capacity. Rate of change in the turbine mass flow rate ($\Delta\dot{m}_T$), and rate of change in the efficiency of the turbine ($\Delta\eta_T$) for each cycle due to erosion can be calculated as:

$$\Delta\dot{m}_T = 1 + (0.5 * EI/(100 * N)) * i \tag{2.6.3}$$

$$\Delta\eta_T = 1 - (EI/(100 * N)) * i \tag{2.6.4}$$

where i denotes the cycle number and N is the total number of cycles that the turbine erosion will be completed in the specific EI. Using the linear relationship erosion index (EI) of 2% and 3% can be calculated based on Table 2.5.

Data is generated from our Simulink model described in Section 2.5 in different

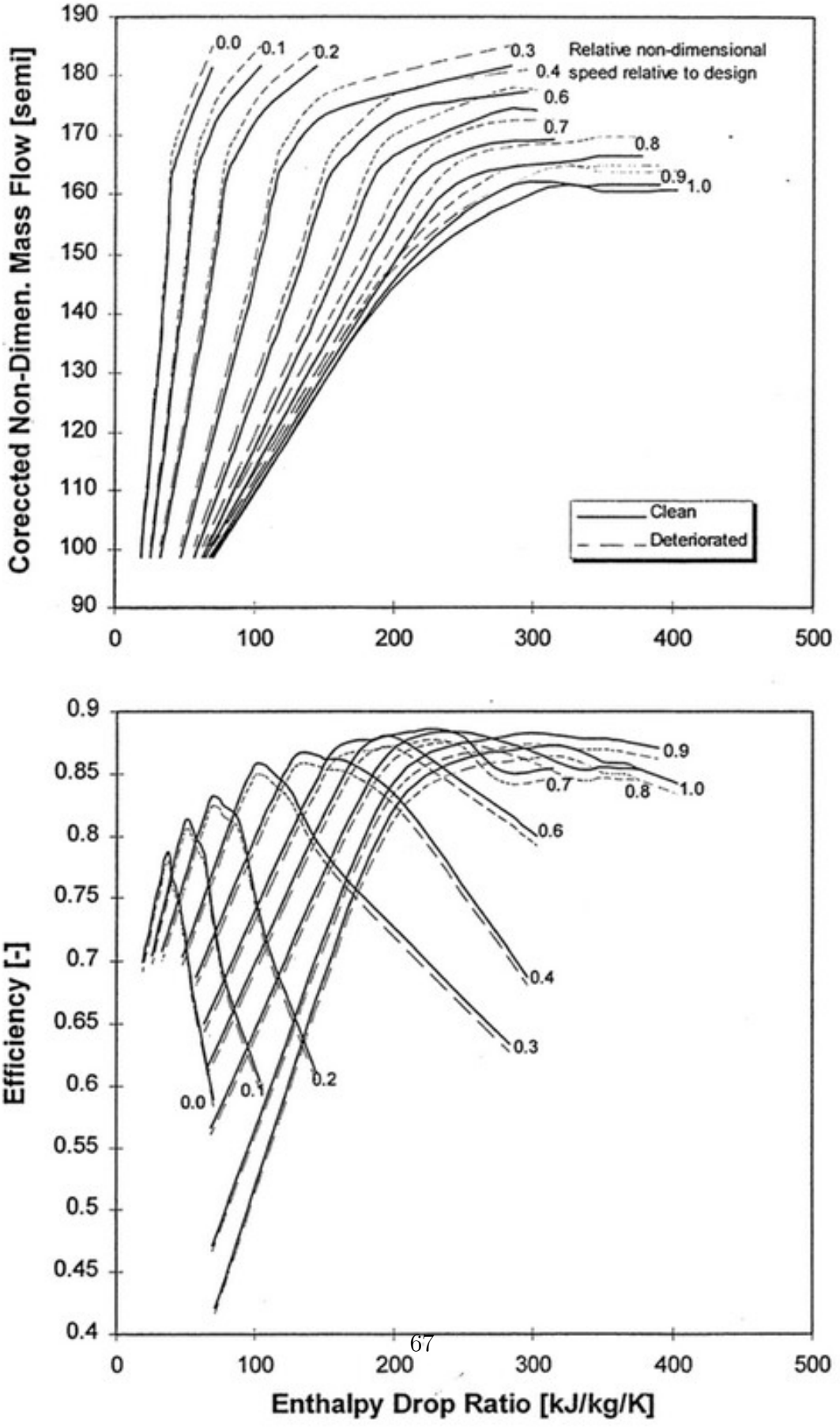

Figure 2.17: Turbine map in presence of erosion [136].

Table 2.5: Linear relationship among EI, efficiency and mass flow rate

Erosion Index	$\Delta\eta_T$ reduction	$\Delta\dot{m}_T$increase
1%	1%	0.5%
2%	2%	1%
3%	3%	1.5%

erosion scenarios for 200 flight cycles. It is also possible for the user to change the erosion cycles for the turbine. Percentage change in each jet engine output measurement for different erosion indices is presented in Table 2.6. ΔT_T denotes the change in the turbine temperature, ΔT_P denotes the change in the turbine pressure, ΔC_T denotes the change in the compressor temperature, ΔW_f denotes the change in the fuel flow, and ΔN denotes the change in the spool speed. The results related to the turbine erosion modelling is depicted in Figures 2.18-2.21. It can be seen that turbine temperature and turbine pressure increase while the compressor output temperature decreases. The value of fuel injected is increased to maintain constant take-off thrust. The spool speed has a decreasing pattern gain. The increases in the erosion index have more effects on the variations of the engine parameters.

Table 2.6: Percent change in each measurement for different erosion indices

Erosion Index	$\Delta\%T_T$	$\Delta\%T_P$	$\Delta\%C_T$	$\Delta\%C_P$	$\Delta\%W_f$	$\Delta\%$N
1%	0.5415	-0.0015	-0.5627	-0.7119	0.4091	-0.7593
2%	1.09	0.0001	-1.1109	-1.4089	0.8273	-1.5115
3%	1.6386	0.003	-1.6466	-2.09	1.236	-2.2497

2.6.5 Concurrent Degradations

It is also possible that both compressor fouling and turbine erosion occur in the gas turbine engine at the same time. This degradation is also modelled in our gas turbine model described in Section 2.5 for different scenarios ($FI = 1\%$ and $EI = 1\%$,

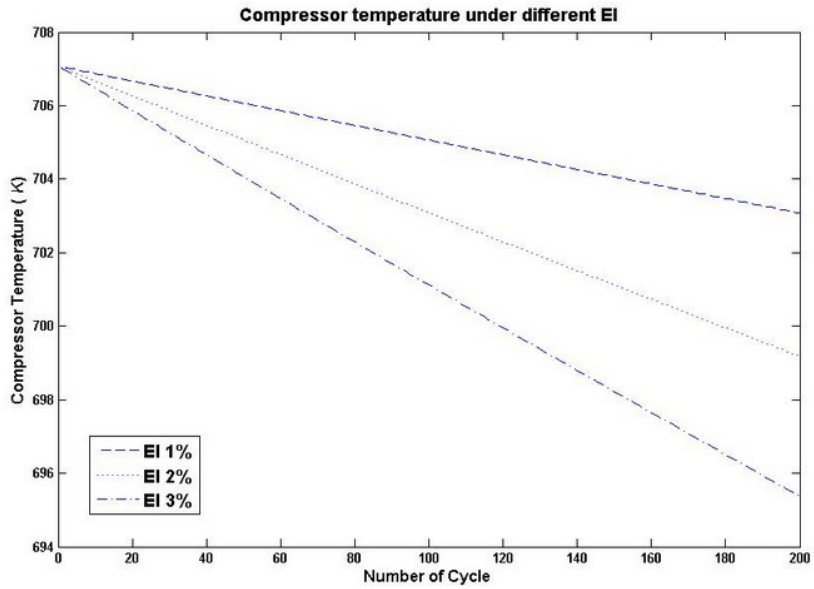

Figure 2.18: Changes in compressor temperature under presence of different EI for the model.

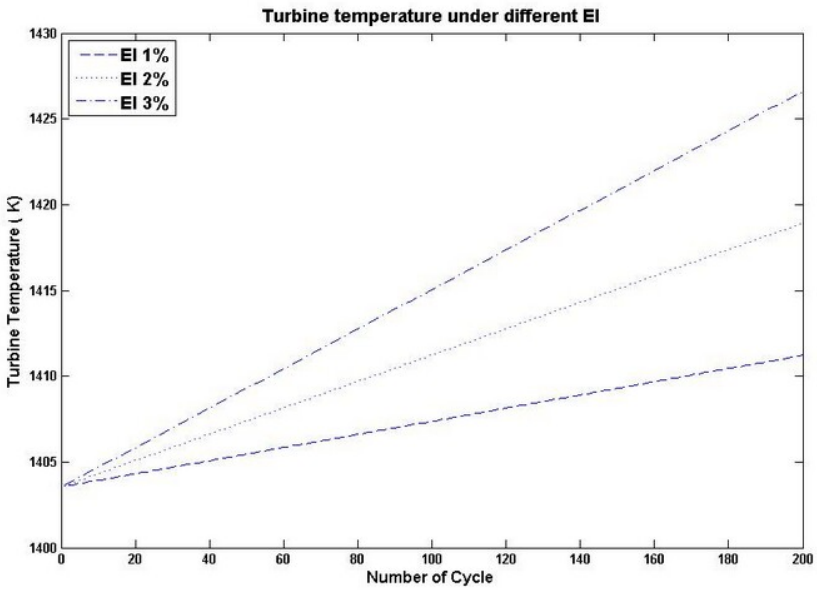

Figure 2.19: Changes in turbine temperature under presence of different EI for the model.

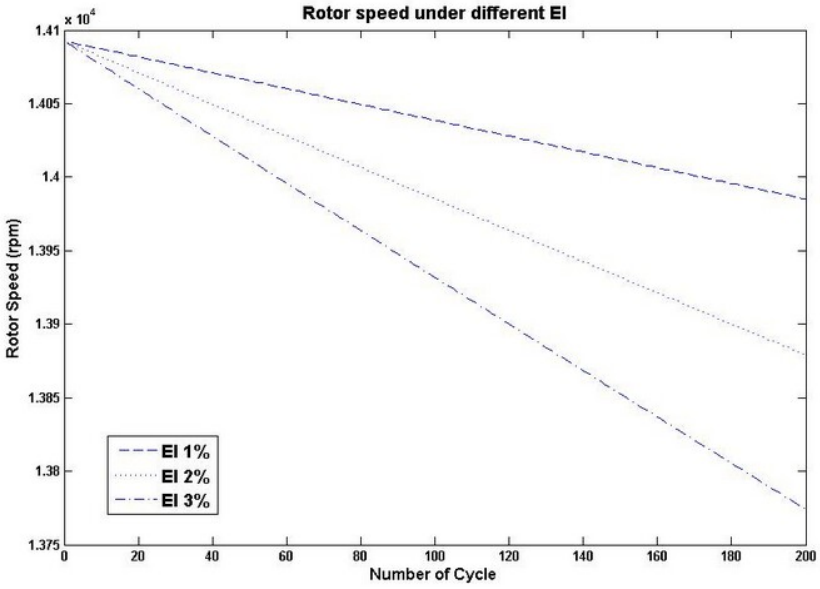

Figure 2.20: Changes in rotor speed under presence of different EI for the model.

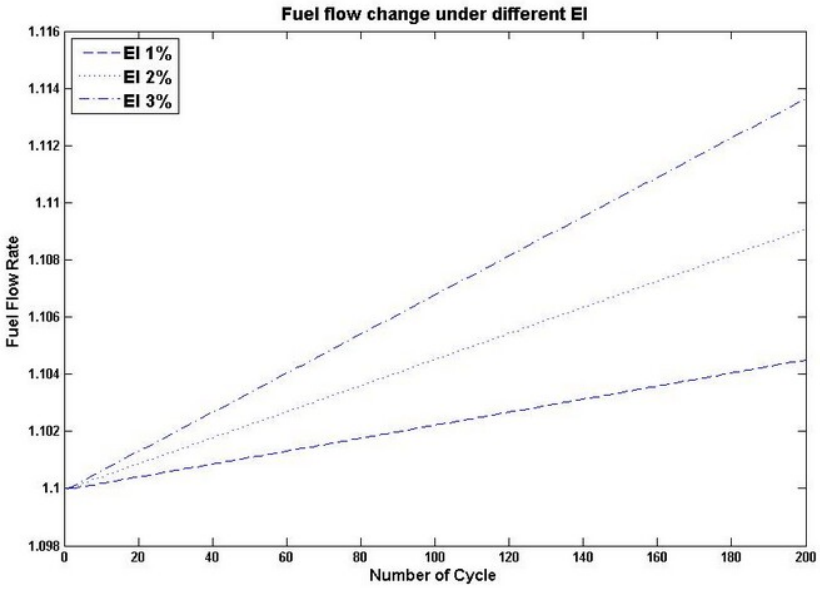

Figure 2.21: Changes in fuel flow rate under presence of different EI for the model.

$FI = 1\%$ and $EI = 2\%$, $FI = 1\%$ and $EI = 3\%$, $FI = 2\%$ and $EI = 1\%$, $FI = 2\%$ and $EI = 2\%$, $FI = 2\%$ and $EI = 3\%$, $FI = 3\%$ and $EI = 1\%$, $FI = 3\%$ and $EI = 2\%$ and $FI = 3\%$ and $EI = 3\%$).

Based on the work of Naeem [142], a linear relationship is considered on the health parameters of the system such that 1% fouling in the compressor decreases the efficiency by 1% and causes a reduction of 0.5% in the compressor flow capacity while at the same time 1% erosion in the turbine decreases the efficiency by 1% and causes an increase of 0.5% in the turbine flow capacity. Using equations (2.6.1)-(2.6.4), fouling and erosion indices denotes by (FEI) can be calculated based on Table 2.7 for the total number of cycles to be 200.

Table 2.7: Linear relationship among FI, EI, η_C, η_T, $\dot{m}_C$ and $\dot{m}_T$

Fouling Index	Erosion Index	η_C reduction	η_T reduction	$\Delta\dot{m}_C$ reduction	$\Delta\dot{m}_T$increase
1%	1%	1%	1%	0.5%	0.5%
1%	2%	1%	2%	0.5%	1%
1%	3%	1%	3%	0.5%	1.5%
2%	1%	2%	1%	1%	0.5%
2%	2%	2%	2%	1%	1%
2%	3%	2%	3%	1%	1.5%
3%	1%	3%	1%	1.5%	0.5%
3%	2%	3%	2%	1.5%	1%
3%	3%	3%	3%	0.5%	0.5%

Percentage change in each jet engine measurement for different fouling and erosion indices is shown in Table 2.8. Measurement changes are also shown in Figures 2.22-2.25.

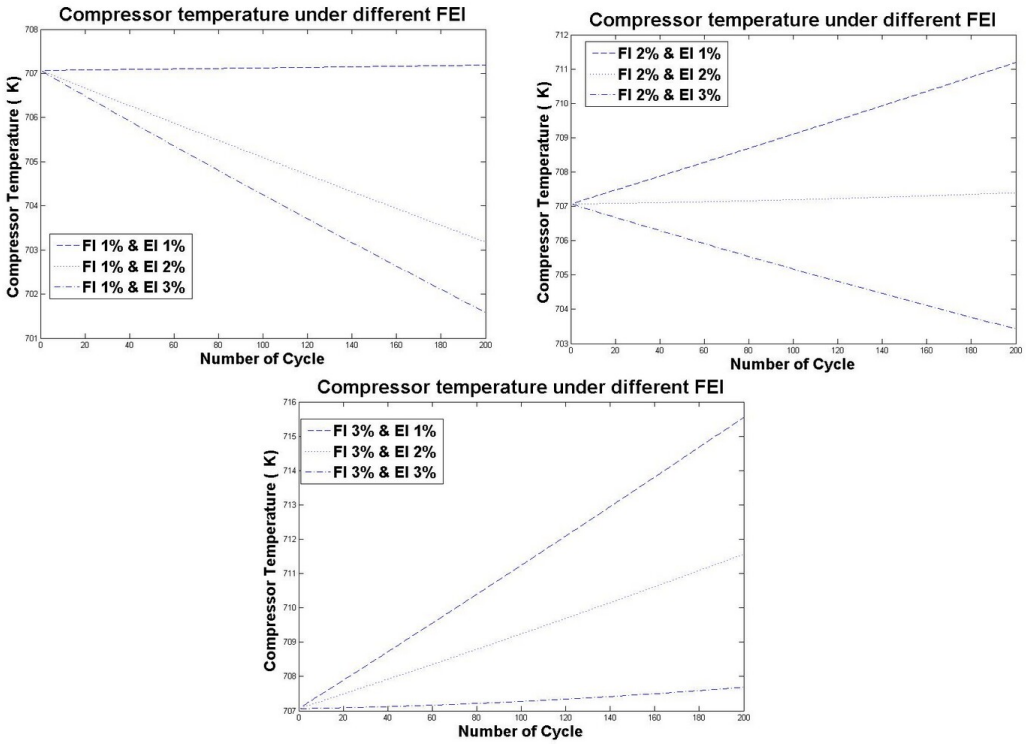

Figure 2.22: Changes in compressor temperature under presence of different fouling and erosion indices for the model.

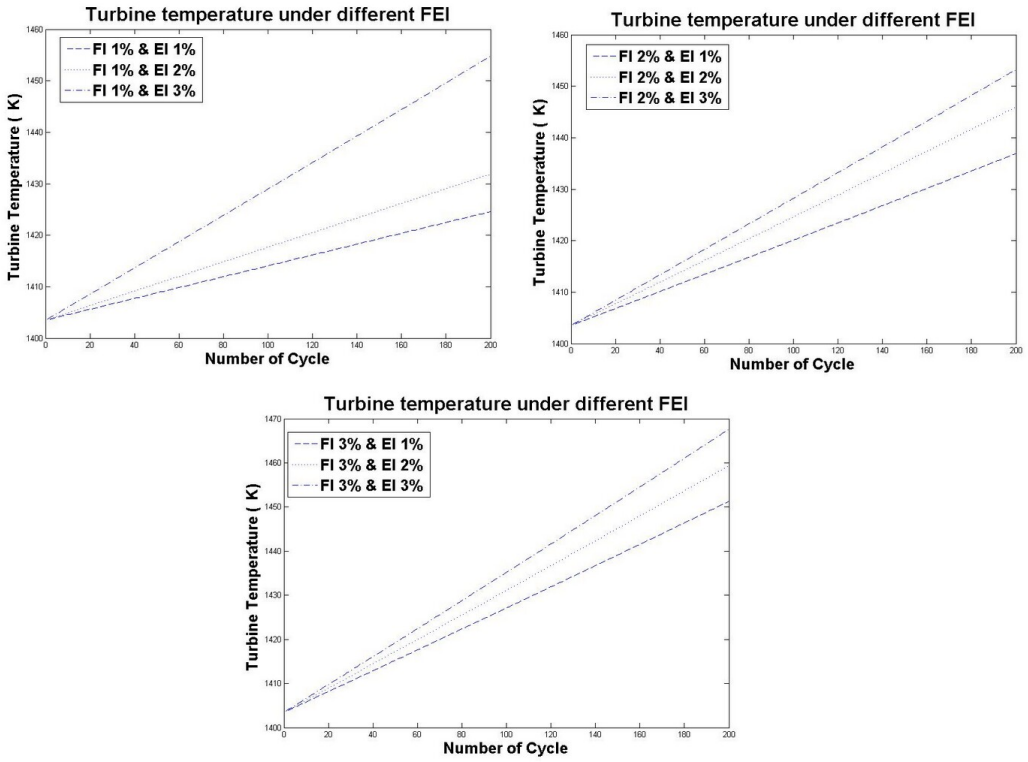

Figure 2.23: Changes in turbine temperature under presence of different fouling and erosion indices for the model.

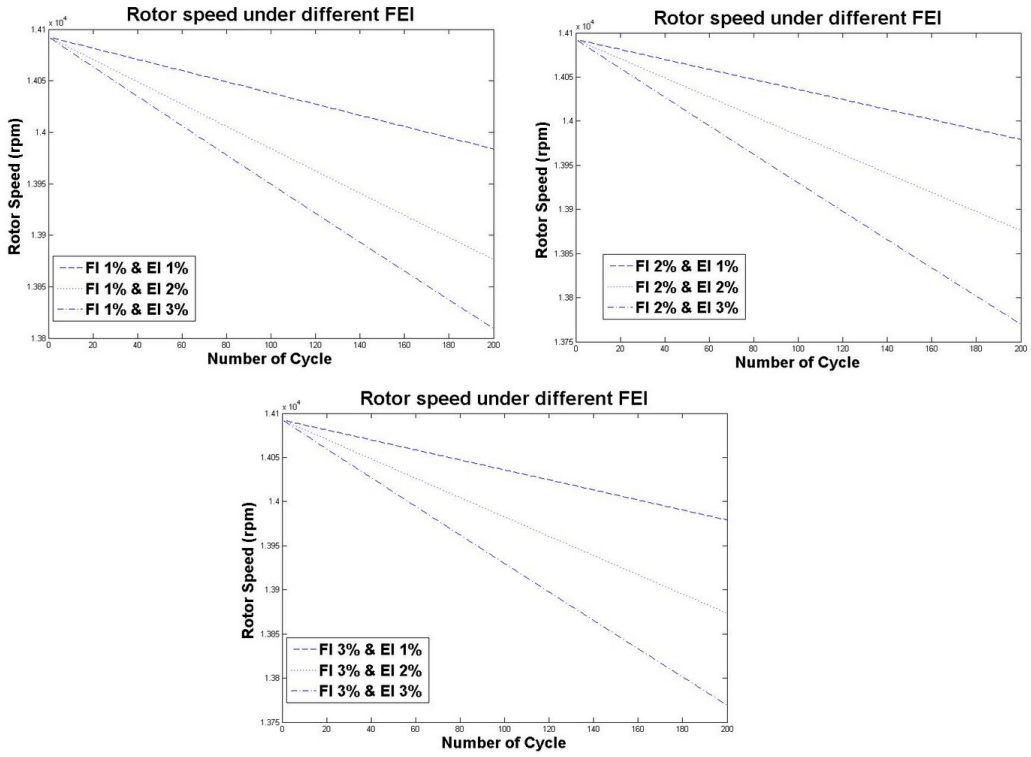

Figure 2.24: Changes in rotor speed under presence of different fouling and erosion indices for the model.

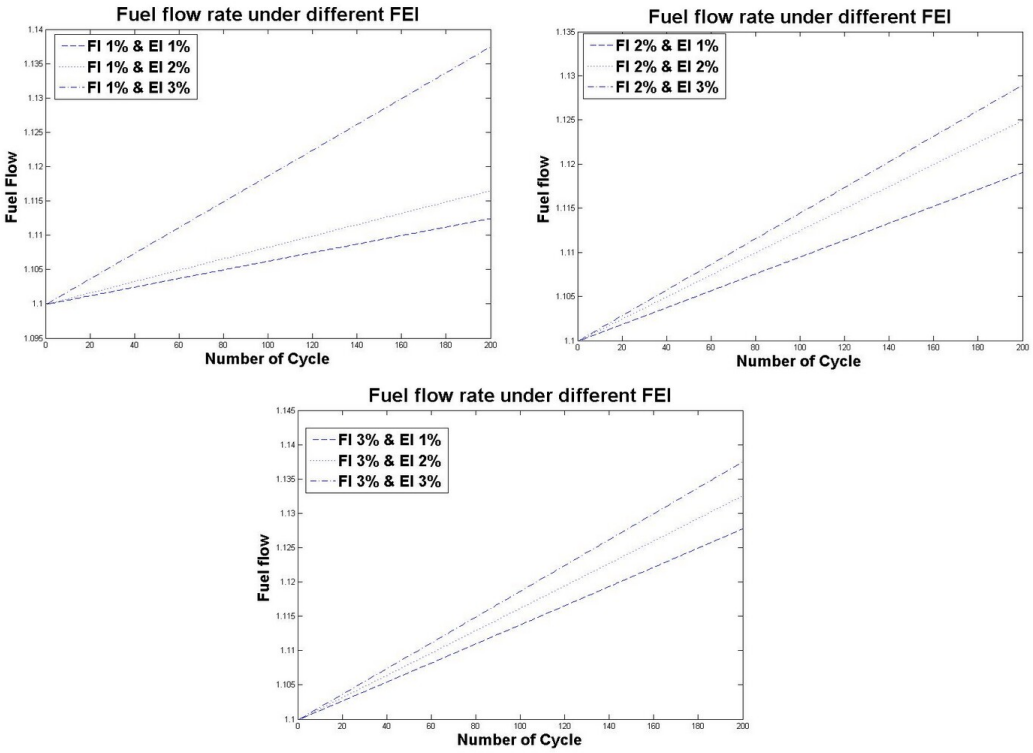

Figure 2.25: Changes in fuel flow rate under presence of different fouling and erosion indices for the model.

Table 2.8: Percent change in each measurement under presence of compressor fouling and turbine erosion at the same time

Fouling Index	Erosion Index	$\Delta\%T_T$	$\Delta\%T_P$	$\Delta\%C_T$	$\Delta\%C_P$	$\Delta\%W_f$	$\Delta\%$N
1%	1%	1.49	0.003	0.017	-0.706	1.127	-0.7663
1%	2%	2.016	-0.0165	-0.5471	-1.423	1.5	-1.525
1%	3%	3.64	0.6199	-0.7714	-1.5041	3.4	-2.001
2%	1%	2.365	-0.0465	0.5798	-0.7525	1.727	-0.8019
2%	2%	3.006	0.0045	-0.0461	-1.401	2.264	-1.5258
2%	3%	3.519	-0.015	-0.512	-2.1077	2.627	-2.278
3%	1%	3.391	-0.0165	1.1948	-0.7224	2.5182	-0.8019
3%	2%	3.961	-0.009	0.6333	-1.4125	2.9545	-1.547
3%	3%	4.5448	0.0075	0.088	-2.0849	3.41	-2.2851

2.7 Gas Turbine Simulation Program (GSP)

2.7.1 Overview

Gas turbine simulation program (GSP) [166] is a component based modelling environment which allows steady-state and transient simulation of any gas turbine configuration and was developed by National Aerospace Laboratory (NLR). This software has been used for various applications such as performance analysis, control system design and diagnosis [164]. It is also used for sensitivity analysis of some variables such as ambient conditions and component degradations. Moreover, flight conditions, degradation and malfunctions of control can be analysed. New engines based on the need of the user can be developed by just dragging and dropping the components and defining their measurements. A model of the engine component is shown in Figure 2.26 [166].

Simulation with GSP is based on modelling processes in different components of the gas turbine with aerodynamic and thermodynamic relations and component maps. It configures different predefined components such as inlet, compressor, combustion chamber, turbine and exhaust nozzles corresponding to the specified gas turbine type which is needed for simulation. Parameters such as air or gas properties, rotor speeds

and efficiencies determine the component operating point. The condition of the gas in a component exit forms the inlet condition for the next component [166]. A set of user specified design data points is defined and non-linear differential equations are determined by the mass balance, heat balance, equation of conservation of momentum and the power balance for all components. In this thesis, GSP is used to validate the

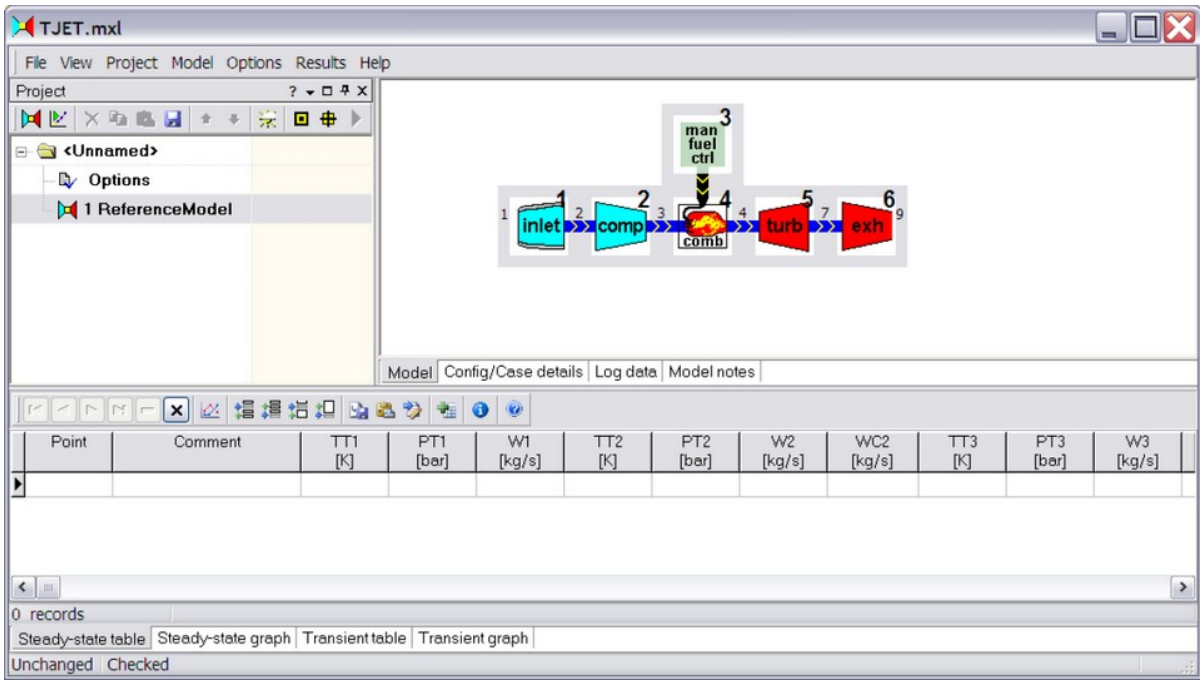

Figure 2.26: A model of engine component in the GSP software [166].

degraded data generated with our jet engine model. A simple turbojet engine model configuration representing an engine similar to the General Electric J85 is used for data validation which is deteriorated with the same degree as a single spool jet engine model that was described earlier in Section 2.5.

2.7.2 Data Generation using the GSP Software

As mentioned earlier, General Electric J85 is used to validate the data for fouling and erosion degradations, and in the case when both fouling and erosion occur at the same time. The results based on the percentage changes in turbine temperature,

compressor temperature, spool speed and fuel flow rate are shown in Tables 2.9-2.11 for different fouling indices, erosion indices, and both fouling and erosion indices.

Table 2.9: Percentage changes in measurements in presence of compressor fouling using GSP

Fouling Index	$\Delta\%T_T$	$\Delta\%C_T$	$\Delta\%N$	$\Delta\%W_f$
1%	1.253	0.4717	-0.3887	1.447
2%	2.304	0.8291	-0.6378	2.368
3%	3.594	1.223	-0.8747	3.684

Table 2.10: Percentage changes in measurements in presence of turbine erosion using GSP

Erosion Index	$\Delta\%T_T$	$\Delta\%C_T$	$\Delta\%N$	$\Delta\%W_f$
1%	1.797	-0.33	-1.221	1.974
2%	3.847	-0.6467	-2.017	3.947
3%	5.911	-0.8641	-2.679	6.053

2.7.3 Data Validation using the GSP Software

GSP is used to validate deteriorated data obtained from a single spool jet engine model given by equations (2.6.1)-(2.6.4). The percentage of changes in the turbine temperature, rotor speed, and fuel flow rate under presence of different compressor fouling indices in our model and the GSP software are compared together in Table 2.12, where it follows that the changes in the measurements follow the same pattern for the model and the GSP software. The turbine temperature increases in both of them while the rotor speed decreases its value due to the fouling indices. The differences between the GSP results and the developed model are due to the differences in the engine parameters. The fuel flow levels are not the same, and the compressor and turbine maps are different.

Table 2.11: Percentage changes in measurements in presence of both compressor fouling and turbine erosion using GSP

Fouling Index	Erosion Index	$\Delta\%T_T$	$\Delta\%C_T$	$\Delta\%N$	$\Delta\%W_f$
1%	1%	3.054	0.0571	-1.415	3.158
1%	2%	5.095	-0.201	-2.089	5.263
1%	3%	7.168	-0.4219	-2.745	7.368
2%	1%	4.369	0.5066	-1.488	4.605
2%	2%	5.744	-0.035	-2.558	5
2%	3%	7.918	-0.208	-3.122	7.368
3%	1%	5.553	0.9231	-1.622	5.71
3%	2%	7.555	0.6725	-2.289	7.684
3%	3%	9.652	0.4459	-2.939	9.789

Table 2.12: Comparing the percentage changes in measurements in presence of compressor fouling for the model and GSP software

FI	$\Delta\%T_T(model)$	$\Delta\%T_T(GSP)$	$\Delta\%N(model)$	$\Delta\%N(GSP)$	$\Delta\%W_f(model)$	$\Delta\%W_f(GSP)$
1%	0.94	1.253	-0.0142	-0.3887	0.7182	1.447
2%	1.888	2.304	-0.0213	-0.6378	1.418	2.368
3%	2.864	3.594	-0.0355	-0.8747	2.15	3.684

Table 2.13 compares the changes in the turbine temperature, rotor speed, and fuel flow rate occurs in the model due to different erosion indices with the same measurements in the GSP software. It is verified from Table 2.13 that the erosion phenomena has the same effect on our model and the model in the GSP software. The comparison between the generated data in the model and data generated from the GSP software when the engine has both fouling in the compressor and erosion in the turbine at the same time with different rates are presented in Table 2.14. Although there are discrepancies between the results of the GSP and the developed model due to different fuel flow levels and compressor and turbine maps, the results from both models show the same trend.

Table 2.13: Comparing the percentage changes in measurements in presence of turbine erosion for the model and GSP software

EI	$\Delta\%T_T(model)$	$\Delta\%T_T(GSP)$	$\Delta\%N(model)$	$\Delta\%N(GSP)$	$\Delta\%W_f(model)$	$\Delta\%W_f(GSP)$
1%	0.5415	1.797	-0.7593	-1.221	0.4091	1.974
2%	1.09	3.847	-1.5115	-2.017	0.8273	3.947
3%	1.6386	5.911	-2.2497	-2.679	1.236	6.053

Table 2.14: Comparing the percentage changes in measurements in presence of both compressor fouling and turbine erosion for the model and GSP software

FI	EI	$\Delta\%T_T(model)$	$\Delta\%T_T(GSP)$	$\Delta\%N(model)$	$\Delta\%N(GSP)$	$\Delta\%W_f(model)$	$\Delta\%W_f(GSP)$
1%	1%	1.49	3.054	-0.7663	-1.415	1.127	3.158
1%	2%	2.016	5.095	-1.525	-2.089	1.5	5.263
1%	3%	3.64	7.168	-2.001	-2.745	3.4	7.368
2%	1%	2.365	4.369	-0.8019	-1.488	1.727	4.605
2%	2%	3.006	5.744	-1.5258	-2.558	2.264	5
2%	3%	3.519	7.918	-2.278	-3.122	2.627	7.368
3%	1%	3.391	5.553	-0.8019	-1.622	2.5182	5.71
3%	2%	3.961	7.555	-1.547	-2.289	2.9545	7.684
3%	3%	4.5448	9.652	-2.2851	-2.939	3.41	9.789

2.8 Conclusion

In this chapter, an overview of the NARX and Elman neural network structures along with their learning algorithms were addressed. Different performance evaluation methods of an estimator were presented. To overcome the problem of uncertainty in measurable variables, prediction intervals were introduced. The aircraft jet engine mathematical model and its equations were introduced as a basis for our data generation. Different gas turbine deteriorations are discussed. Two important ones, namely compressor fouling and turbine erosion are modelled in our jet engine Simulink model and deteriorated data are validated by using the GSP software. It is also considered that both compressor fouling and turbine erosion occur with different rates at the same time. These data are also validated by using the GSP software.

Chapter 3

Jet Engine Prediction using NARX Neural Networks

The goal of this chapter is to predict turbine temperature for some flights ahead for engine's maintenance actions. Using the predicted output from the NARX neural network, one is able to decide whether the temperature exceeds a threshold at specific flight or the next flights will be safe. The objective of the NARX neural network is to predict dynamics of the degradation in the engine. Various simulations are carried out in this chapter to demonstrate the performance of the NARX neural network in terms of prediction horizons.

Data generated from our model which was described in Section 2.6 in presence of compressor fouling, turbine erosion and in the case when both of these degradations occur are used to train and test the neural networks. As described in Section 2.1.1, the NARX neural network composes of three layers of input, hidden and output layers. Fuel flow rate is used as an input data and the turbine output temperature is the output of the NARX neural network. Turbine temperature increases in presence of fouling or erosion degradations. However, it should be noted that fouling and erosion occur in multiple flights and they do not change the engine measurements severely in

only one flight. Therefore, samples which are used to train the network are obtained from multiple flights. These samples represent the maximum values of fuel flow rate and turbine temperature in each flight.

The NARX neural network uses the training data to represent the non-linear model of the engine. The input (fuel flow rate) is given to the NARX neural network. The neural network then processes the input and compares the network output against the actual outputs. Errors (the difference between the network output and actual output) are then propagated back through the system, causing the system to adjust the weights. This process occurs over and over as the weights are continually adjusted. During the training step of the neural network the same set of data is processed many times as the connection weights are refined. The process stops if a pre-specified criterion is fulfilled, e.g. if all the absolute partial derivatives of the error function with respect to weights ($\frac{\partial E}{\partial w}$) are smaller than a given threshold. Architecture for the NARX neural network during the training step is shown in Figure 3.1. Simulated data in Figure 3.1 were obtained in Section 2.6. The number of input and output delays are shown as d_u and d_y.

The best NARX neural network found in the training step is now evaluated using the testing data sets. These data sets differ from the training data sets and they are given as an unseen data to the neural network. Fuel flow rate is fed to the NARX neural network as an input and the turbine temperature is predicted as the output of the network. The schematic of the NARX neural network approach during testing phase can be seen in Figure 3.2 where d_u is the number of input delays and d_y is the number of output delays.

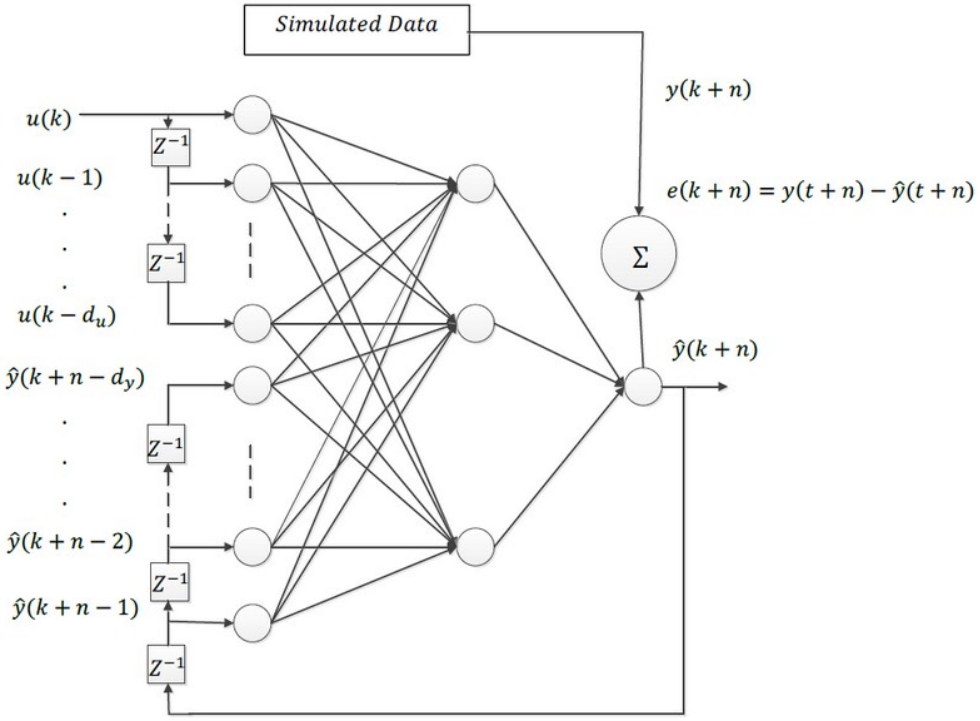

Figure 3.1: Architecture for the NARX neural network during the training phase.

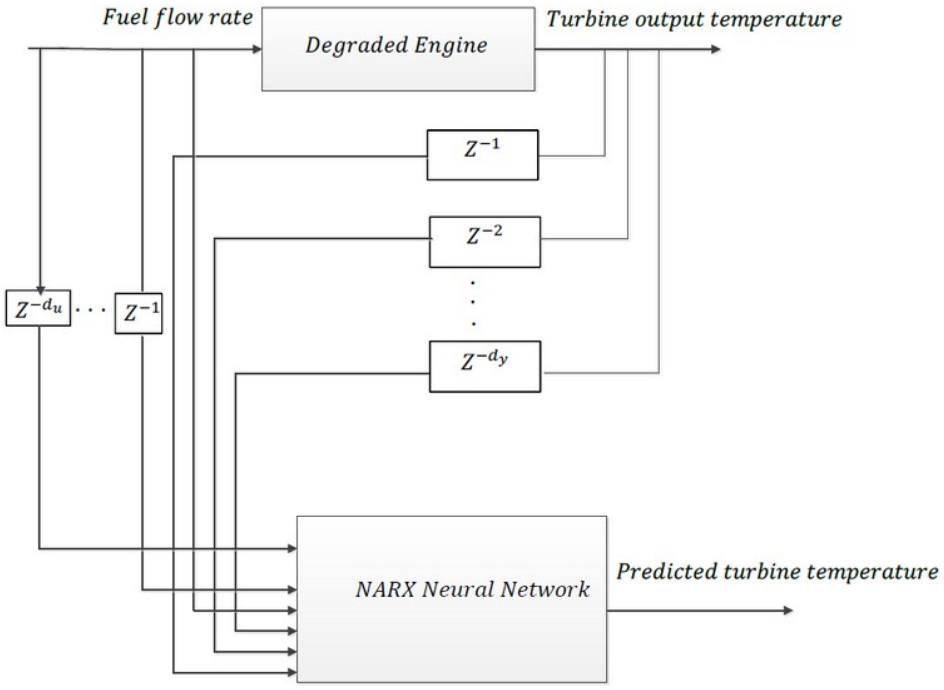

Figure 3.2: Architecture of the NARX neural network during the testing phase.

3.1 Simulation Results

There are different measures which affect the performance of the NARX neural network namely; the number of hidden neurons, the size of training data set, the number of input delays and the number of output delays. Small networks with small number of hidden neurons cannot learn the dynamics of the system accurately while large networks tend to over fit the training data [161]. Optimal NARX neural network structure is achieved by using different number of training data sets to predict 2 flights ahead turbine output temperature. We will also use 5 steps, 8 steps, and 12 steps ahead turbine temperature to evaluate the applicability of the NARX neural network in long term prediction.

3.1.1 Compressor Fouling

As mentioned in Section 2.6.2, compressor fouling consumes up to 80% of the total engine performance loss during operation. This deterioration decreases the compressor mass flow rate and efficiency which causes power reduction and an increase in the heat rate. Losses due to fouling can be compensated by compressor washing or surface cleaning. This fact necessitate the importance of knowing the appropriate time for compressor washing. The objective of this section is to predict the influence of compressor fouling on gas turbine performance for maintenance actions. Different NARX neural network structures are trained by using different number of training data. These networks are then used to predict turbine output temperature for multi flights ahead. Turbine output temperature is predicted in presence of different amounts of compressor fouling. The predicted value is then used for maintenance actions to decide whether the next flights will be safe or not.

3.1.1.1 FI = 1%

The number of neurons in the hidden layer plays an important role in the performance of the network. In the present section, the optimum number of hidden neurons is achieved through trial and error procedure. The delayed version of fuel flow rate and turbine temperature are given as inputs to train the network. The parameters d_u and d_y are both set to 3 and the 2 flights ahead turbine temperature is predicted using networks with different number of hidden neurons. The entire data set equals to 200 points which implies that the compressor fouled in 200 simultaneous flights by the amount of 1%. A total of 80 data points are used for training and 120 for the neural network evaluation. The statistical error measures such as standard deviation, mean and RMSE for prediction are presented in Table 3.1 for different network structures.

Table 3.1: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.5776	4.0479	4.7847
6	**2.6167**	**2.3338**	**3.4998**
7	1.0639	3.3499	3.5014
8	3.8568	2.1575	4.4148
9	3.1553	2.4520	3.9897
10	3.1845	2.8004	4.2330
11	2.1527	3.3640	3.9820
12	3.7566	2.5407	4.5291
13	3.4892	3.8537	5.1867
14	3.8795	4.5680	5.9786
15	5.2047	4.1398	6.6396

Based on Table 3.1, the best NARX network performance based on the RMSE is achieved when the network has 6 hidden neurons. Turbine temperature data used for training the network structure of 7 inputs (current and 3 previous values of the

fuel flow rate as well as 3 previous values of the turbine temperature), 6 hidden neurons and 1 output (current turbine temperature) as well as the actual and predicted values are depicted in Figure 3.3. The prediction errors which are the absolute difference between the actual and predicted values are shown in Figure 3.4. Based on the information given in Section 2.4 about uncertainty in measurements and prediction bounds, actual and predicted data with their prediction bounds based on 95% probability are shown in Figure 3.5. It can be seen that only 58.33% data are within the prediction horizons.

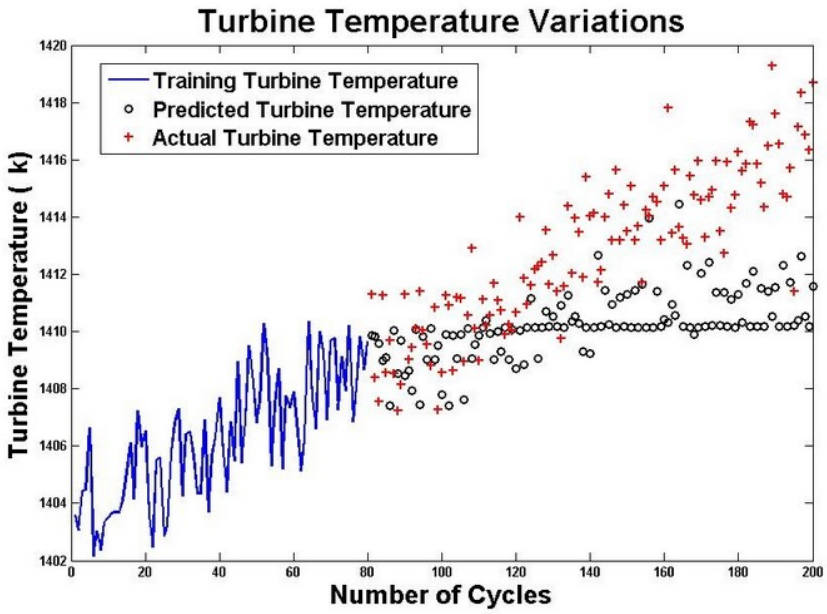

Figure 3.3: Turbine temperature variations subject to $FI = 1\%$ using NARX 7-6-1 during training and testing phases.

Next the number of the training data are increased to 60% of the entire data set which implies that 120 data points are used in the training phase and 80 are used in the testing phase. The number of the hidden neurons increases from 5 to 15 and turbine output temperatures are predicted for the 2 flights ahead. The results are shown in Table 3.2 where the network with the structure of 7-5-1 has the lowest error. Figure 3.6 shows that 68.75% of the predicted data are within the prediction intervals.

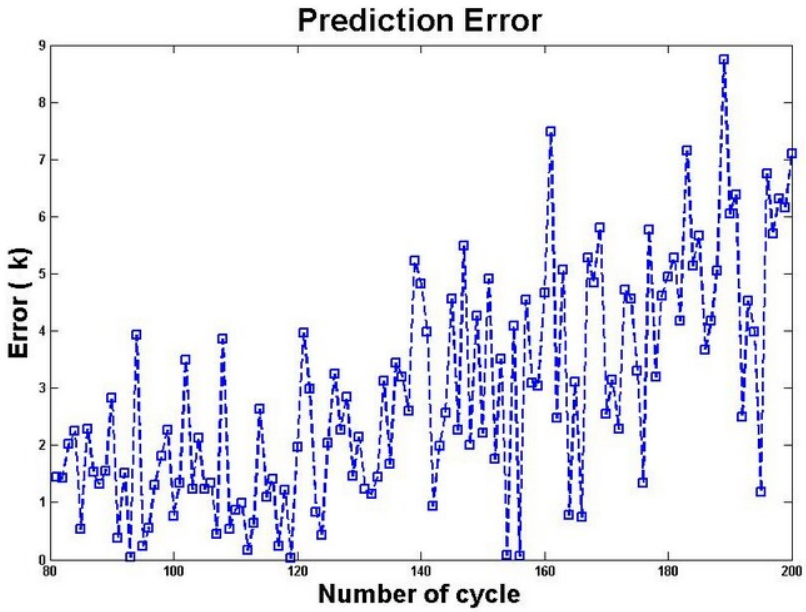

Figure 3.4: Prediction errors for the 2 step ahead turbine temperature when $FI = 1\%$ using NARX 7-6-1 trained with 40% of the available data.

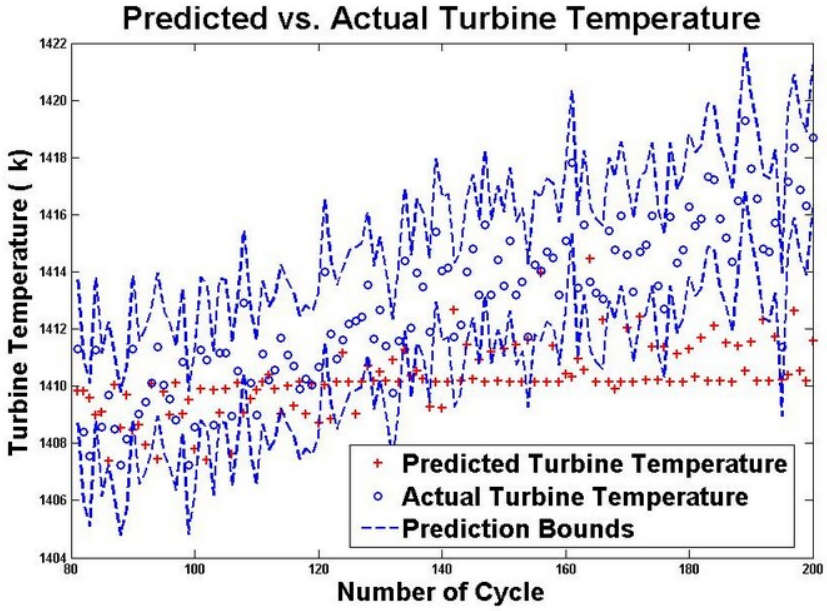

Figure 3.5: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 40% of the available data for $FI = 1\%$.

Table 3.2: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	**1.9805**	**1.8456**	**2.6993**
6	2.8032	1.6829	3.2641
7	3.5124	1.9249	3.9995
8	2.6224	1.8953	3.2286
9	3.8344	1.9537	4.2979
10	3.9653	1.9776	4.4256
11	2.8956	3.1396	4.2566
12	3.5776	2.8405	4.5571
13	3.8389	2.1842	4.4100
14	4.2684	2.0062	4.7110
15	4.8514	2.2619	5.3468

In order to investigate the effect of the training data sets on the performance of the NARX neural network prediction, the training data is increased to 80% of the available data points. Hence 160 data are used in the training phase and 40 ones are used to predict the turbine temperature in the testing phase. The optimal network structure in this case is the network with 6 hidden neurons as shown in Table 3.3. The actual and predicted values are shown pointwise in Figure 3.7 where 100% of the predicted data are within the uncertainty bounds. Based on Tables 3.1-3.3, prediction error decreases 51.66% as the number of the training data increase because the network can learn the dynamics of the degradation better in presence of more data. However, it is not always possible to have much data available.

In the next scenario, the turbine temperature is predicted for 5 flights ahead where 40% of the entire data points are used in the training phase and the rest (120 data points) are used to evaluate the performance of the network. The optimal NARX neural network structure based on Table 3.4 is 7-7-1. The actual and predicted values are shown in Figure 3.8 where 60.83% of the predicted data points are within

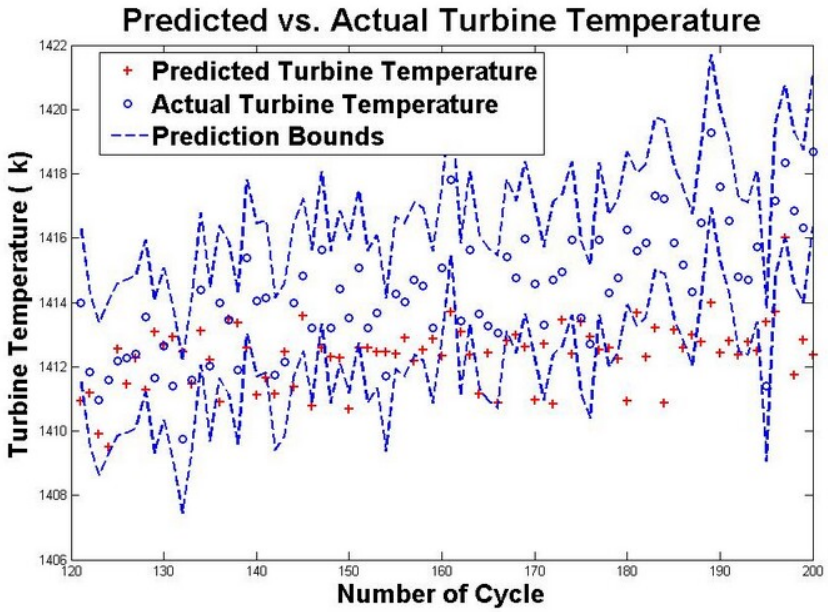

Figure 3.6: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-5-1 trained with 60% of the available data for $FI = 1\%$.

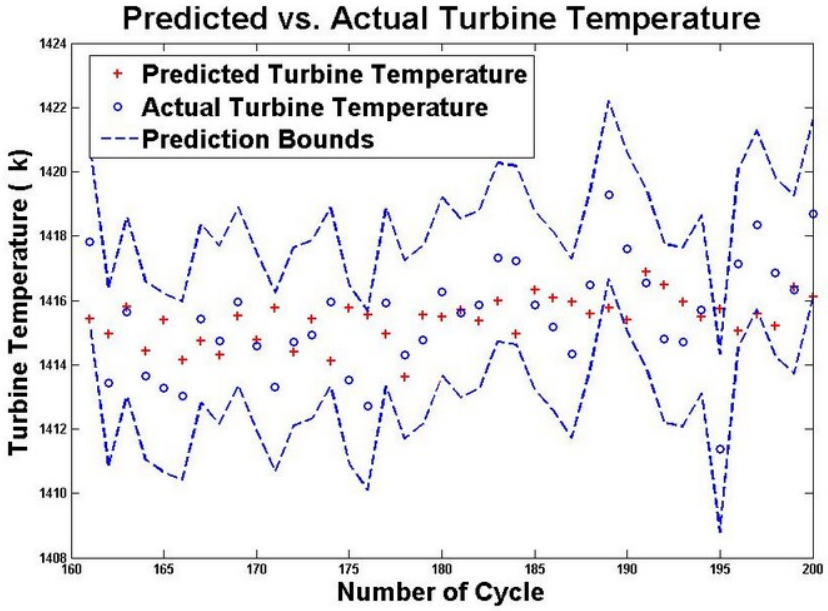

Figure 3.7: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 with 80% training data for $FI = 1\%$.

Table 3.3: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	1.8102	1.7120	2.4768
6	**0.0766**	**1.7117**	**1.6919**
7	1.7986	1.8276	2.5478
8	1.9416	1.7004	2.5669
9	1.9871	1.8543	2.7021
10	2.4426	1.8386	3.0434
11	1.4774	1.6930	3.5505
12	1.4774	3.7634	3.9990
13	1.7884	3.6690	4.0403
14	3.6927	1.8780	4.1321
15	4.5816	1.9018	4.9515

the upper and the lower prediction bounds.

When the NARX neural network is trained with 60% of the entire data points available (the total number of data points are 200), the neural network with 8 hidden neurons has the lowest RMSE as presented in Table 3.5. Actual and predicted turbine temperatures are depicted in Figure 3.9 where 70% of the predicted data points are within the prediction intervals.

Different NARX neural network structures are trained using 160 data points in the training phase and 40 data points in the testing phase to appreciate the importance of the number of training data in the neural network performance. The results of the prediction error are compared together in Table 3.6. The NARX neural network with the structure 7-6-1 has the lowest RMSE. The prediction mean, standard deviation and RMSE are $1.5425K$, $1.8375K$, and $2.3814K$, respectively. The network prediction temperatures along with their actual values are shown in Figure 3.10 where 77.5% of the predicted data points are within the prediction bounds.

The applicability of the NARX neural network to predict turbine temperature in

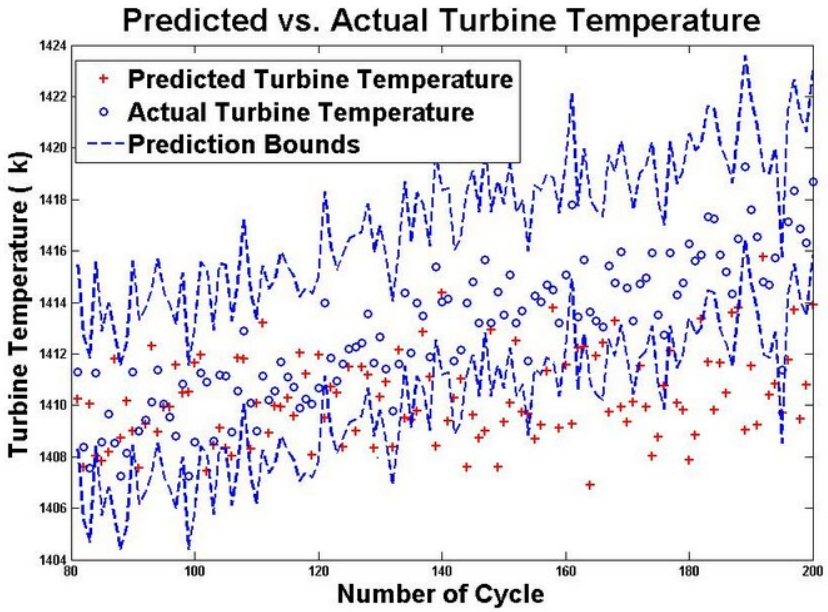

Figure 3.8: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 40% of the available data for $FI = 1\%$.

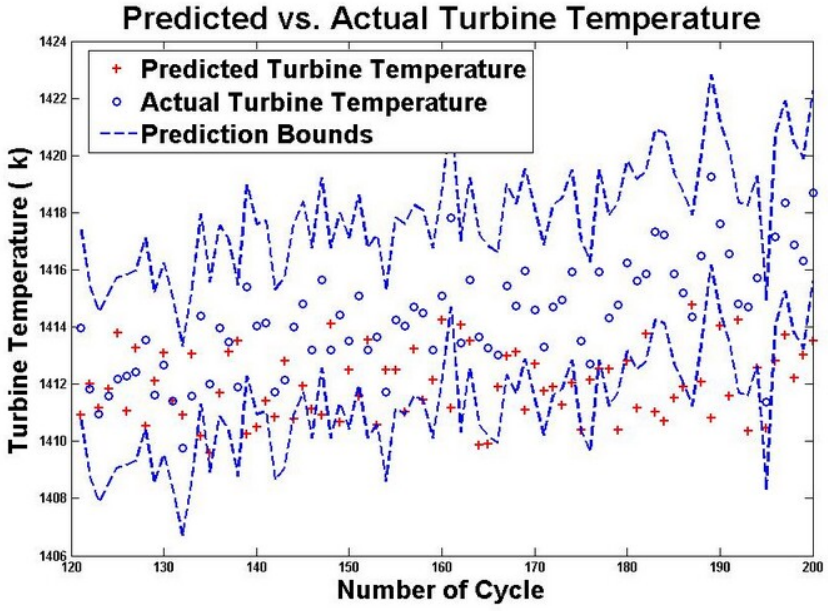

Figure 3.9: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $FI = 1\%$.

Table 3.4: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.6253	2.6311	4.4730
6	3.8449	2.6745	4.6772
7	**2.5449**	**2.8564**	**3.8167**
8	1.9008	3.3641	3.8518
9	2.2717	3.2920	3.9885
10	3.0889	3.0349	4.3215
11	3.8741	2.4695	4.5887
12	3.1836	3.5238	4.7380
13	0.3031	4.9366	4.9253
14	0.3100	5.1798	5.1675
15	4.1248	3.1686	5.1933

8 flights ahead under presence of 1% fouling is investigated next. The training data points are increased from 40% of the entire data points to 80%. The number of hidden neurons are changed from 5 to 15, and the optimal structure is found. The results of the RMSE for different neural network structures when 80 data points are used in the training phase and 120 data points are used in the testing phase are presented in Table 3.7. The NARX neural network with 8 hidden neurons has the lowest RMSE. Based on Figure 3.11, 62.5% of the predicted data points are between the upper and the lower prediction bounds.

Training data is increased to 120 data points which is equal to 60% of the entire data points. The RMSE is in its lowest value when the number of hidden neurons is 8 as shown in Table 3.8. To overcome the problem of uncertainty in measurements, the upper and the lower prediction bounds as well as the actual and predicted values are depicted in Figure 3.12 where 70% of the predicted data points are between these bounds.

Next 80% of the total data available are used to train different NARX neural

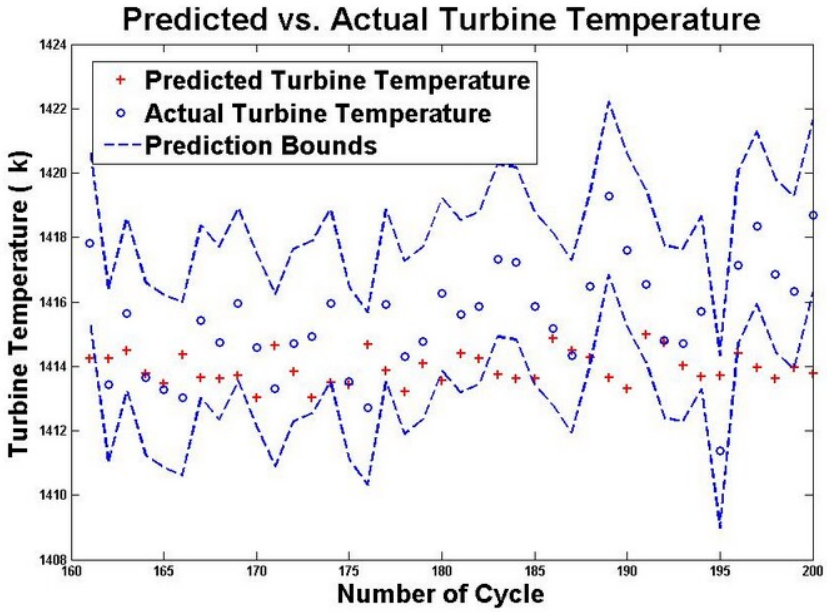

Figure 3.10: The 5 step ahead predicted/actual turbine temperature along with prediction intervals for $FI = 1\%$ using NARX 7-6-1 trained with 80% of the available data.

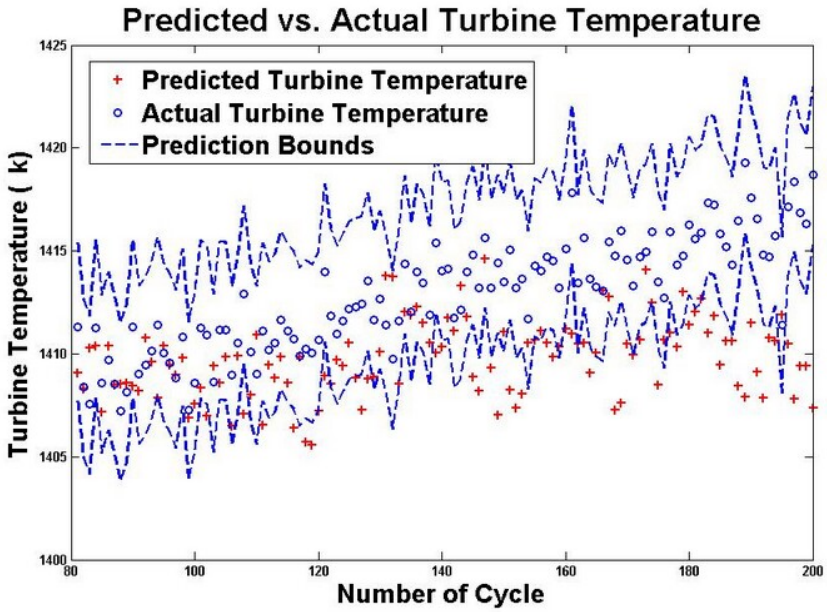

Figure 3.11: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 40% of the available data for $FI = 1\%$.

Table 3.5: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.2785	3.0153	3.7643
6	3.3338	2.2221	3.9987
7	2.7428	1.9469	3.3564
8	**2.3537**	**2.1743**	**3.1950**
9	2.6239	2.8880	3.8885
10	3.2110	2.2843	3.9323
11	3.7704	2.0467	4.2840
12	3.7552	2.0049	4.2510
13	3.8709	2.2521	4.4713
14	3.4111	3.4917	4.8657
15	3.0963	3.8656	4.9339

network structures. These networks are then evaluated to predict turbine output temperatures in 8 flights ahead. The number of hidden neurons are increased from 5 to 15. The results of prediction error based on standard deviation, mean and RMSE are tabulated in Table 3.9. The training and testing data as well as predicted values are shown in Figure 3.13. Prediction bounds are depicted in Figure 3.14 which shows that only 17.5% of the predicted data are outside the bounds. Comparing Tables 3.7 and 3.9, the RMSE decreases by 34.4% when the training data increase from 80 to 160.

Next, various NARX neural networks are trained with different numbers of the training data sets to predict turbine output temperatures in 12 flights ahead. The number of hidden neurons are changed from 5 to 15 and the optimal neural network structure is found. The results of the prediction error for different structures trained with 40% of the available data are shown in Table 3.10. Table 3.11 presented these error values when the networks are trained by 60% of the available data points, and finally the NARX neural networks which are trained by 80% of the entire data points

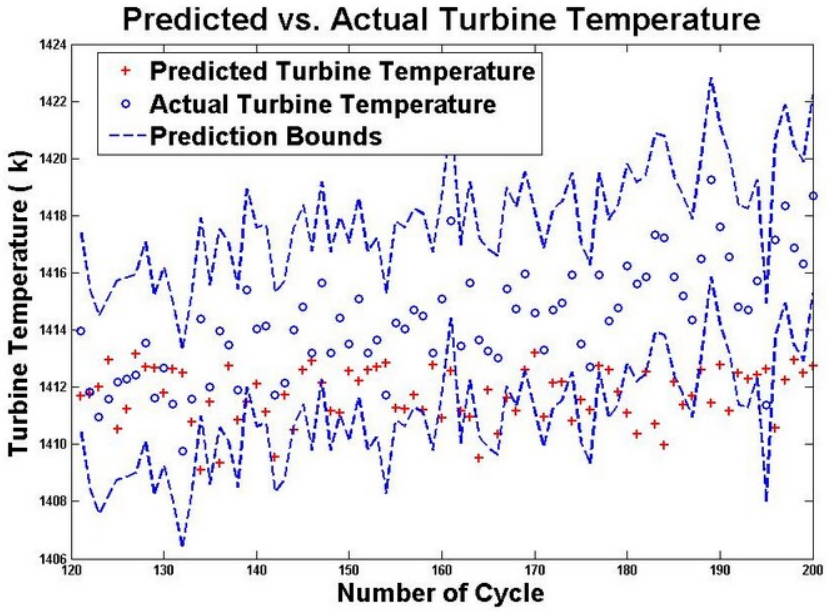

Figure 3.12: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $FI = 1\%$.

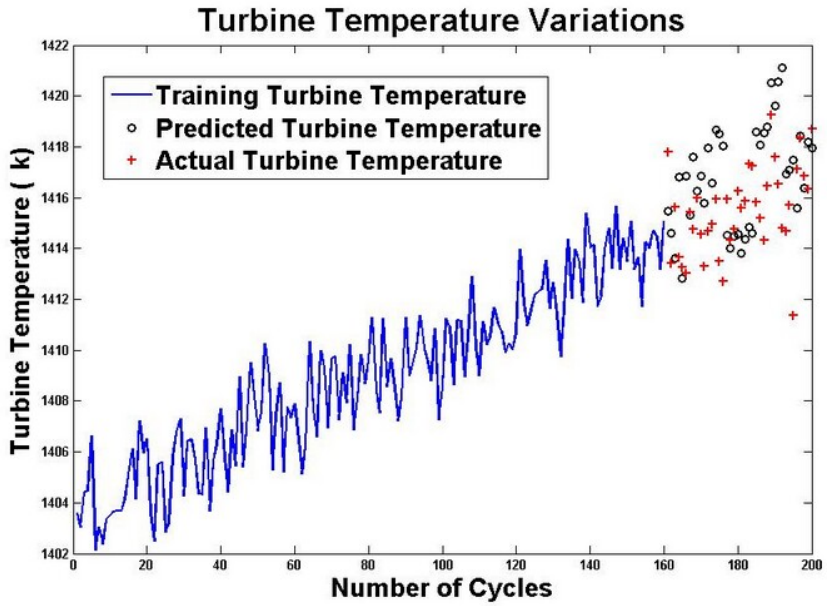

Figure 3.13: Turbine temperature variations subject to $FI = 1\%$ using NARX 7-6-1 trained with 80% of the available data.

Table 3.6: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.0346	2.0921	2.8994
6	**1.5425**	**1.8375**	**2.3814**
7	2.2388	1.9809	2.9730
8	1.6294	2.2107	2.7240
9	1.4532	2.7018	3.0380
10	1.9683	2.3320	3.0293
11	2.4535	2.2731	3.3253
12	2.0097	2.7552	3.3823
13	2.6984	2.0637	3.3814
14	2.8519	2.5030	3.7738
15	2.4999	3.0320	3.9003

are tabulated in Table 3.12.

Actual turbine temperatures and predicted values along with their prediction bounds for these three scenarios are depicted in Figures 3.15-3.17 where 57.5% of the predicted data points are within the prediction bounds when the network is trained with 80 data points, while this value increases to 61.1% for the network trained by using 120 data points and for the neural network trained with 160 data points, 88.89% of the predicted turbine temperatures are between the prediction intervals.

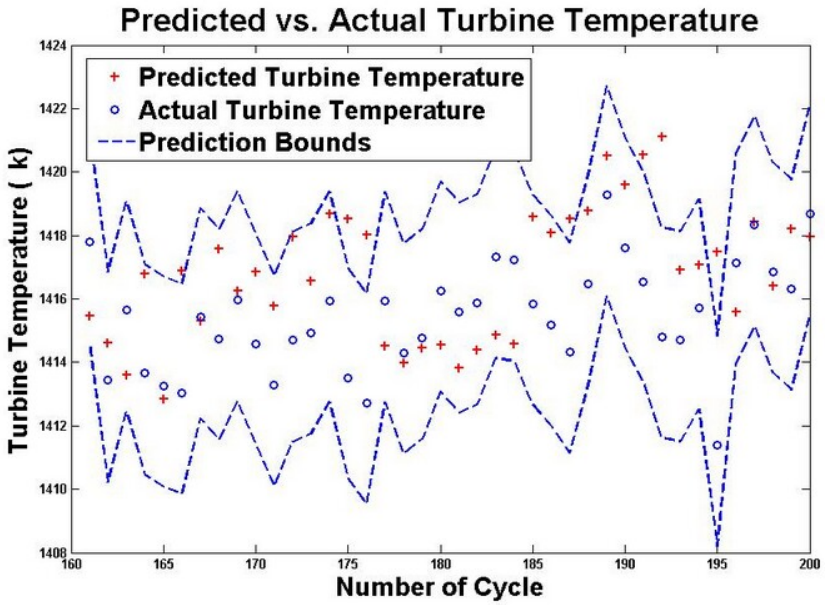

Figure 3.14: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 80% of the available data for $FI = 1\%$.

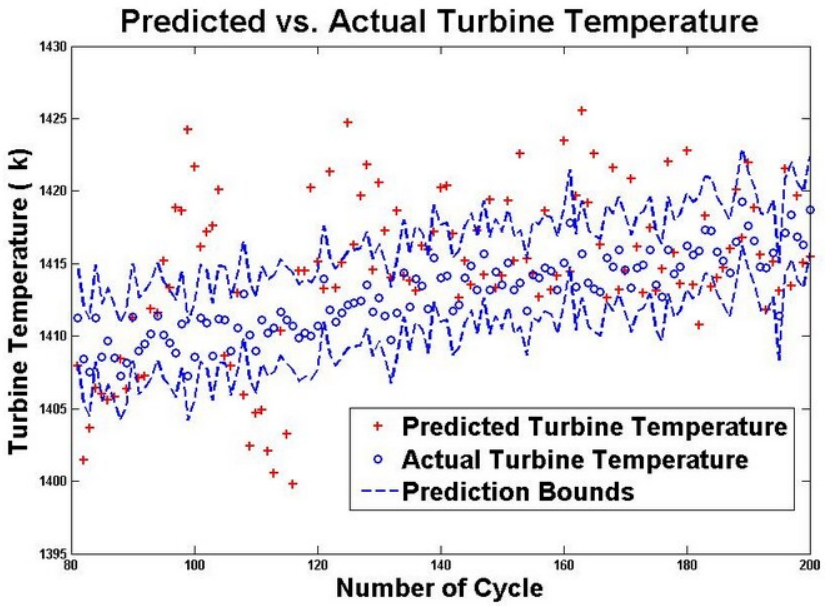

Figure 3.15: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $FI = 1\%$.

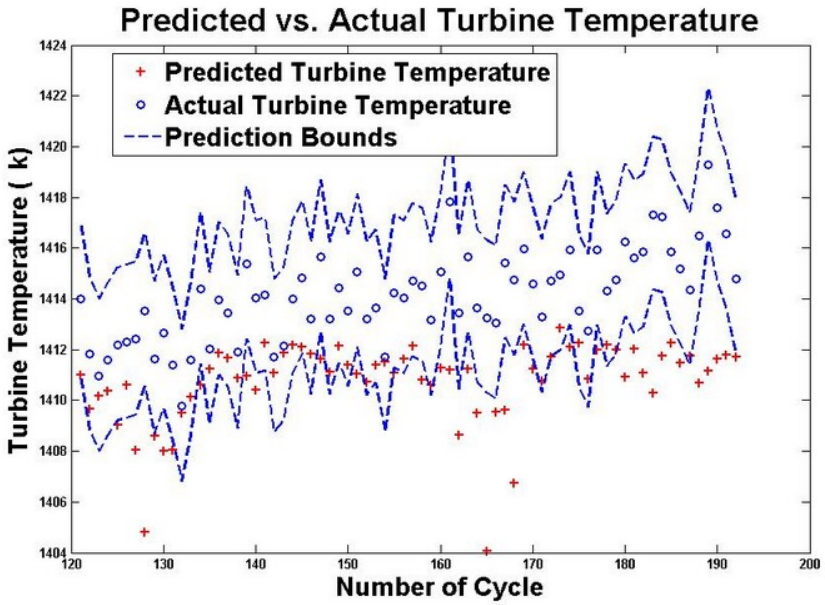

Figure 3.16: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $FI = 1\%$.

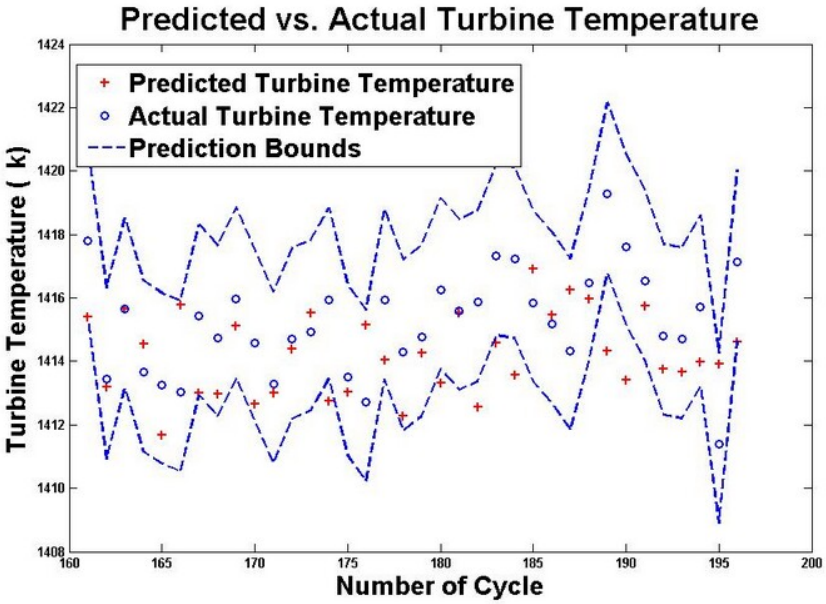

Figure 3.17: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $FI = 1\%$.

Table 3.7: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.1382	2.9569	5.0789
6	1.7743	4.2614	4.5996
7	3.7401	2.5197	4.5039
8	**3.1322**	**2.8074**	**4.1984**
9	2.9408	3.2020	4.3377
10	4.1616	3.1239	5.1959
11	4.1617	3.4311	5.3846
12	2.6271	3.6853	4.5133
13	3.8226	3.8841	5.4381
14	4.4415	3.6370	5.7310
15	5.2705	3.6038	6.3764

Table 3.8: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.2571	2.0085	3.8200
6	3.3481	2.1793	3.9874
7	2.7839	2.5344	3.7540
8	**2.5998**	**2.1558**	**3.3688**
9	2.4317	2.4739	3.4579
10	2.5380	2.4221	3.4978
11	3.1943	2.1783	3.8587
12	3.1258	2.5280	4.0102
13	3.4249	2.2985	4.1167
14	4.1286	2.7753	4.9651
15	2.8730	2.8712	4.0491

Table 3.9: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.4977	1.9023	3.1252
6	**1.2881**	**2.4668**	**2.7554**
7	2.3751	2.0078	3.0938
8	2.8172	2.0319	3.4586
9	1.5381	2.9969	3.3351
10	3.2764	2.0119	3.8316
11	2.7606	2.2603	3.5500
12	3.1816	2.9062	4.2846
13	3.5454	2.3107	4.2161
14	4.3036	2.1944	4.8183
15	3.4992	2.6587	4.3745

Table 3.10: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.2919	3.3382	5.4288
6	4.0675	3.4194	5.3046
7	4.7767	2.8910	5.5772
8	5.2378	3.3058	6.1864
9	**1.6955**	**5.0388**	**5.2964**
10	4.3920	3.0701	5.3513
11	5.3481	3.2862	6.2698
12	4.9234	3.8479	6.2388
13	5.9948	3.4370	6.9031
14	5.5541	3.4922	6.5530
15	2.1726	5.8325	6.2012

Table 3.11: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.2480	2.3616	4.8523
6	4.3784	1.8617	4.7527
7	3.6987	1.8718	4.1395
8	**3.3699**	**1.9627**	**3.8929**
9	3.5993	1.9915	4.1069
10	4.1184	2.1639	4.6453
11	3.4507	2.2633	4.1181
12	3.7096	2.6581	4.5528
13	4.1402	2.4493	4.8018
14	4.5844	2.0549	5.0181
15	4.6346	2.5029	5.2590

Table 3.12: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.5353	1.6491	3.8913
6	3.3528	1.6699	3.7353
7	**2.7459**	**1.6224**	**3.1779**
8	2.8034	2.0116	3.4341
9	3.0484	2.0274	3.6454
10	3.5175	2.4122	4.2462
11	2.5928	3.3978	4.2363
12	4.1541	1.7233	4.4882
13	3.7071	2.6592	4.5406
14	3.4059	3.5026	4.8505
15	4.3935	2.5493	5.0617

3.1.1.2 FI = 3%

In this section, the applicability of the NARX neural network in turbine temperature prediction in presence of 3% compressor fouling is investigated. As mentioned in Section 2.6.2, compressor on-line/off-line washing and cleaning the surface is needed to remove particles when the compressor fouled by the amount of 3%. Data are generated in our Simulink model which was described in Section 2.5 when it goes through 3% compressor fouling in 200 simultaneous flights using equations (2.6.1) and (2.6.2).

As done previously in Section 3.1.1.1, 40% of the entire data sets are used to train different NARX neural network structures to find the optimal numebr of hidden neurons where d_u and d_y are both set to 3 and the 2 flights ahead turbine temperatures are predicted. The results of the prediction error for various NARX neural network structures are tabulated in Table 3.13.

Table 3.13: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.0688	5.1788	5.5567
6	0.6657	4.8422	4.8677
7	1.0134	6.0869	6.1456
8	**0.8400**	**4.3568**	**4.4192**
9	0.4268	4.4224	4.4246
10	1.0309	5.6584	5.7283
11	2.8234	5.8226	6.4492
12	2.5164	4.7866	5.3901
13	3.1873	3.3098	4.5850
14	1.0166	6.7152	6.7640
15	6.0783	4.1862	7.3704

Based on Table 3.13, the NARX neural network with 8 hidden neurons has the

lowest RMSE equal to $4.4192K$. The actual and predicted values along with prediction intervals for the network 7-8-1 are shown in Figure 3.18 where 65.83% of the data used in the testing phase are within the upper and the lower bounds. Errors which are the absolute difference between the actual and the predicted data are shown in Figure 3.19.

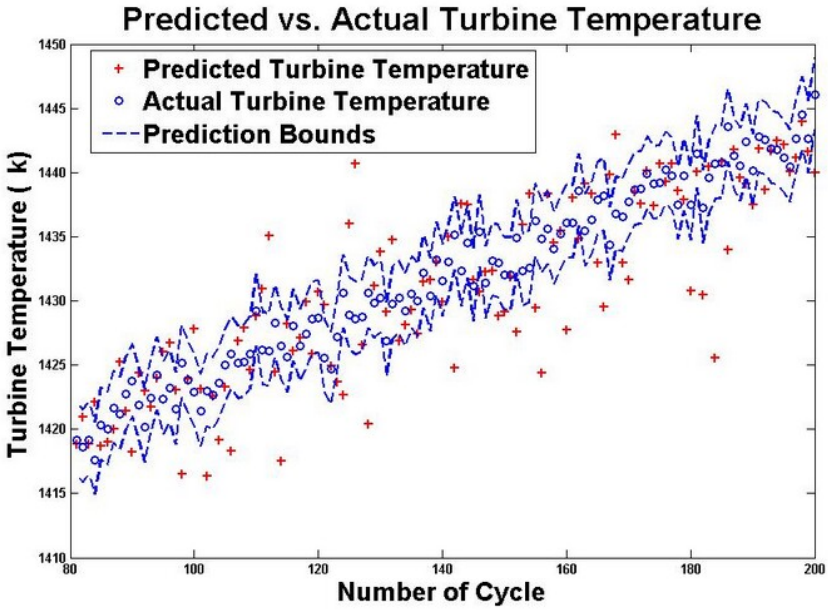

Figure 3.18: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 40% of the available data for $FI = 3\%$.

The training data are increased from 80 to 120 data points and the optimal neural network structure is found by changing the number of hidden neurons from 5 to 15. The networks are tested with 80 points and their prediction errors and standard deviations are presented in Table 3.14. The network with 9 hidden neurons has the best performance in the evaluation phase. The mean, standard deviation and RMSE of the NARX neural network 7-9-1 are $0.4547K$, $3.1614K$, and $3.1743K$, respectively. Figure 3.20 shows that 73.75% of the predicted data are within the prediction intervals.

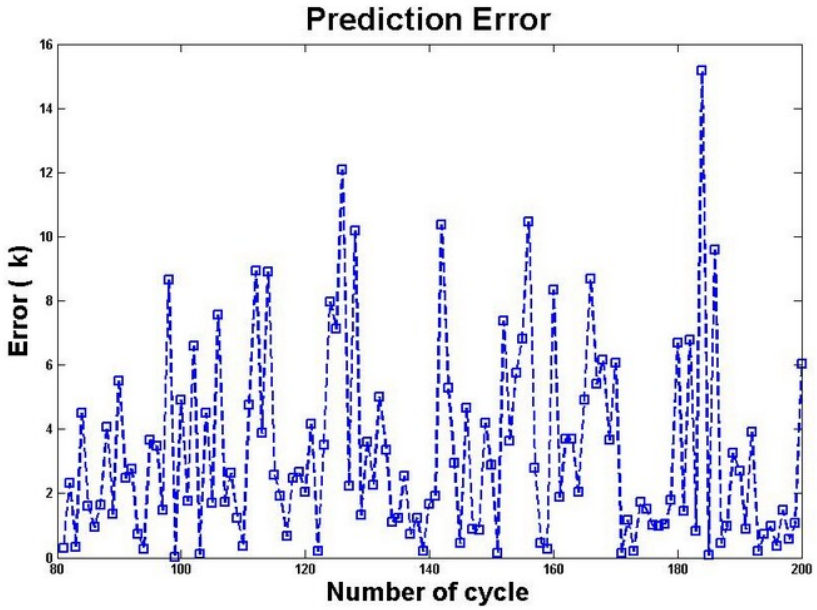

Figure 3.19: Prediction errors for the 2 step ahead turbine temperature when $FI = 3\%$ using NARX 7-8-1 trained with 40% of the available data.

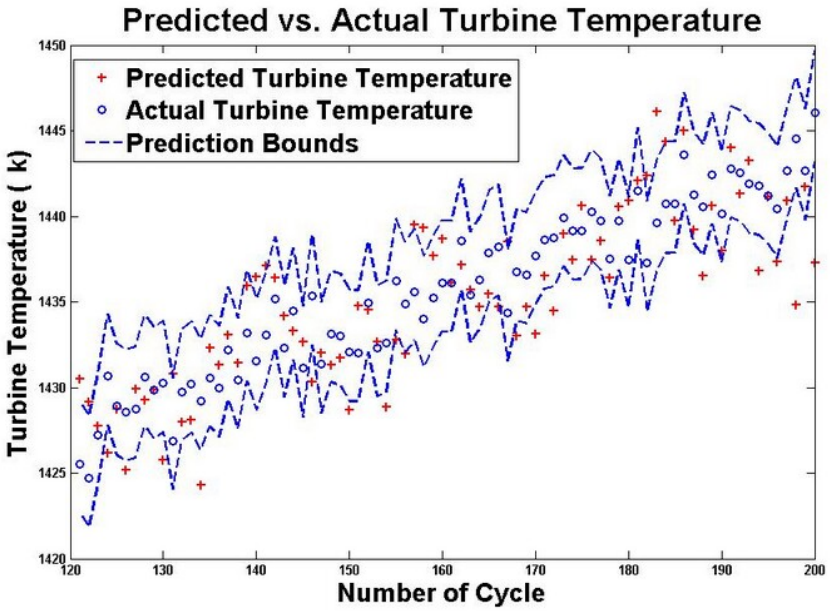

Figure 3.20: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 60% of the available data for $FI = 3\%$.

Table 3.14: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.1705	2.6127	4.0979
6	4.3240	3.3497	5.4568
7	4.4128	2.7910	5.2120
8	2.5080	2.4461	3.4927
9	**0.4547**	**3.1614**	**3.1743**
10	2.0396	2.6107	3.3001
11	2.5544	3.1661	4.0527
12	3.1128	3.1794	4.4353
13	4.4353	3.6198	5.2581
14	4.3715	3.0387	5.3130
15	4.2163	4.1316	5.8851

By increasing the number of training data to 160 (80% of the entire data), the optimum NARX neural structure is 7-8-1. The RMSE decreases 52.3% in comparison to the first simulation when the number of training data was 80 points. The results for various NARX neural network structures are presented in Table 3.15, and the actual and predicted values are depicted in Figure 3.21 where 95% of the data in the testing phase are within the prediction bounds.

In the following step of this section, the optimal NARX neural network structures to predict 5 flights ahead are investigated using different number of training and testing data. Table 3.16 summarizes the prediction error when different network structures are trained with 80 data points and tested with 120 points. Figure 3.22 shows the actual versus predicted values for the network with the structure of 7-9-1 where 53.33% of the predicted turbine temperatures are between the upper and the lower prediction bounds.

Next 120 data points are used to train the networks with different number of hidden neurons. The hidden neurons are increased from 5 to 15. These networks

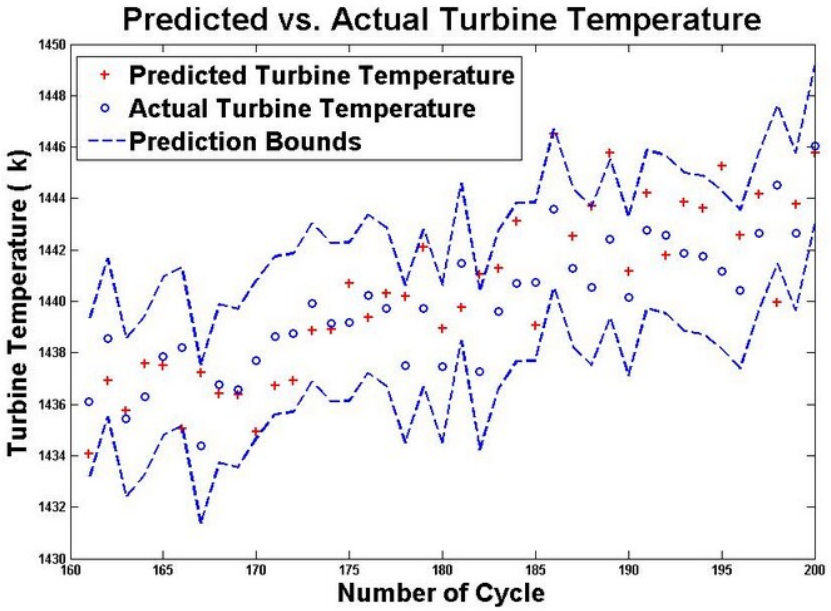

Figure 3.21: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 80% of the available data for $FI = 3\%$.

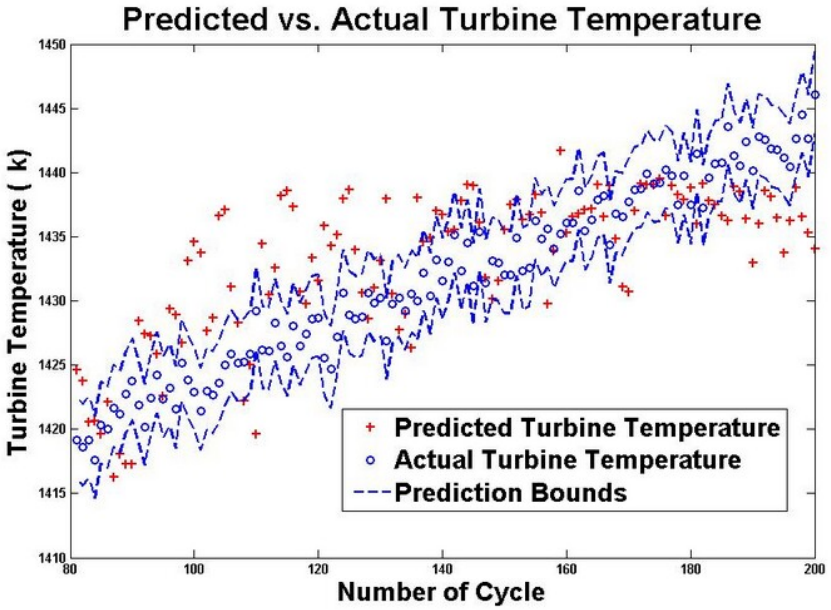

Figure 3.22: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $FI = 3\%$.

Table 3.15: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.4159	1.7286	2.9581
6	1.5032	1.9125	2.4137
7	1.4603	1.7820	2.2866
8	**0.5347**	**2.0649**	**2.1079**
9	2.1451	1.9295	2.8690
10	2.2574	1.8583	2.9090
11	2.3295	2.5747	3.4482
12	2.5831	2.5786	3.6270
13	2.1115	3.1610	3.7683
14	2.0911	3.1281	3.7300
15	2.3277	3.2158	3.9371

are then evaluated by using 80 data points. The results of error in the testing phase are shown in Table 3.17 where the RMSE for the turbine temperature prediction using the network 7-6-1 is $3.4816K$. Actual and predicted data points are depicted in Figure 3.23 where 61.25% of the data used during evaluation are within the prediction intervals. Figure 3.24 shows the errors between the predicted and the actual values.

In order to investigate the effect of the training data points, 160 data are used to train the network. The remaining 40 data points are used in the testing process. The results are summarized in Table 3.18. Based on Figure 3.25, 85% of the predicted data are within the prediction bounds.

The turbine output temperature in 8 flights ahead is predicted using the NARX neural networks. The entire data points are 200 which implies that the compressor degrades by the fouling index of 3% in 200 simultaneous flights. In the first step, the networks are trained by using 40% of the entire data points and the remaining data points which are 120 are used to evaluate the performance of the network in prediction. Then, the training data points are increased to 60% and 80%, respectively.

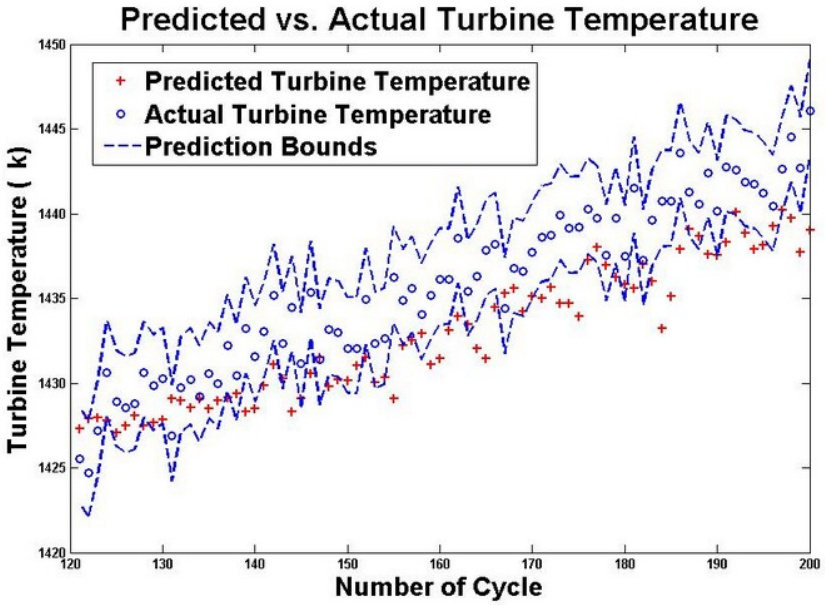

Figure 3.23: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 60% of the available data for $FI = 3\%$.

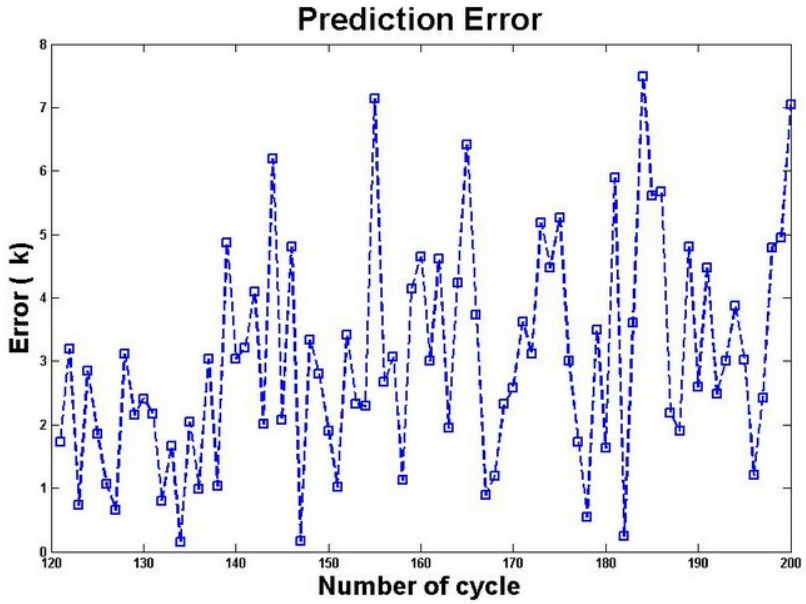

Figure 3.24: The prediction errors for the 2 step ahead turbine temperature when $FI = 1\%$ using NARX 7-6-1 trained with 60% of the available data.

Table 3.16: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.1209	5.6042	6.9374
6	4.6288	4.8678	6.7025
7	1.4835	5.5276	5.7009
8	4.3872	4.7035	6.4177
9	**1.4089**	**5.3722**	**5.5322**
10	4.5696	3.8165	5.9436
11	1.7605	6.2375	6.4562
12	2.8514	5.2357	5.9426
13	1.3867	6.5712	6.6891
14	4.8805	5.0801	7.0293
15	1.7506	6.9985	7.1858

The results of the error in the testing phase are shown in Tables 3.19-3.21. The RMSE decreases 40.5% when the training data increase from 40% to 60% and 49.9% when the training data increased to 80%.

The results of the network which was trained with 40% of the available data points versus the actual values are shown in Figure 3.26 where only 46.66% of the predicted data are within the prediction bounds. This value increases to 72.5% and 77.5% when the networks were trained by using 60% and 80% of the available data points, respectively.

Next the 12 steps ahead turbine temperatures are predicted with available data points in presence of 3% compressor fouling where 80 data sets which is equal to 40% of the entire data are used to train the NARX neural network. The optimal network structure is found by increasing the number of hidden neurons from 5 to 15. Based on Table 3.22, the best performance is achieved with the NARX neural network structure of 7-9-1. Figure 3.29 shows the actual and predicted values. Only 35% of the predicted data are between the upper and the lower prediction bounds.

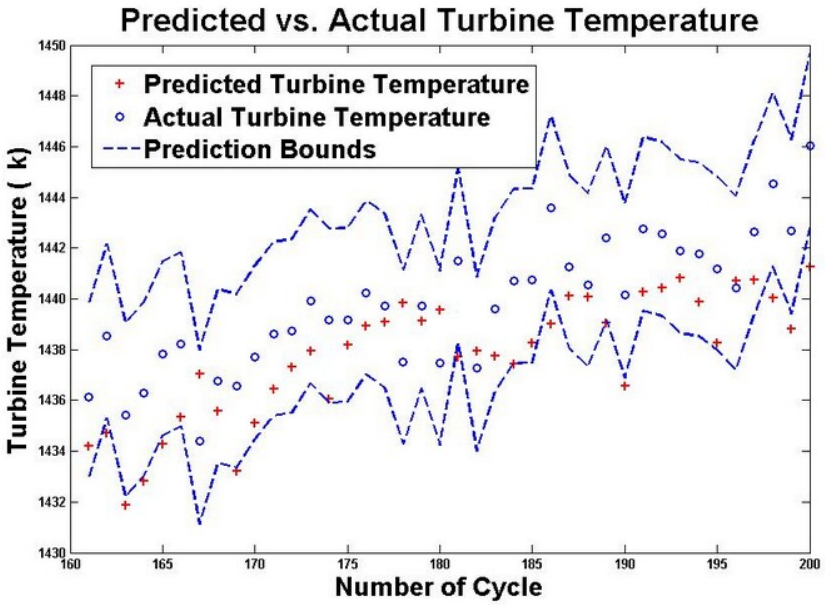

Figure 3.25: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 80% of the available data for $FI = 3\%$.

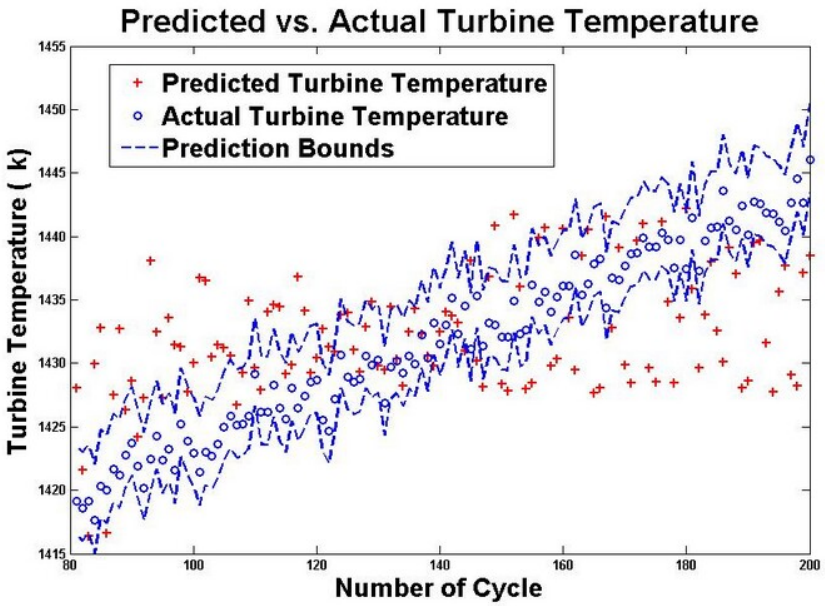

Figure 3.26: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $FI = 3\%$.

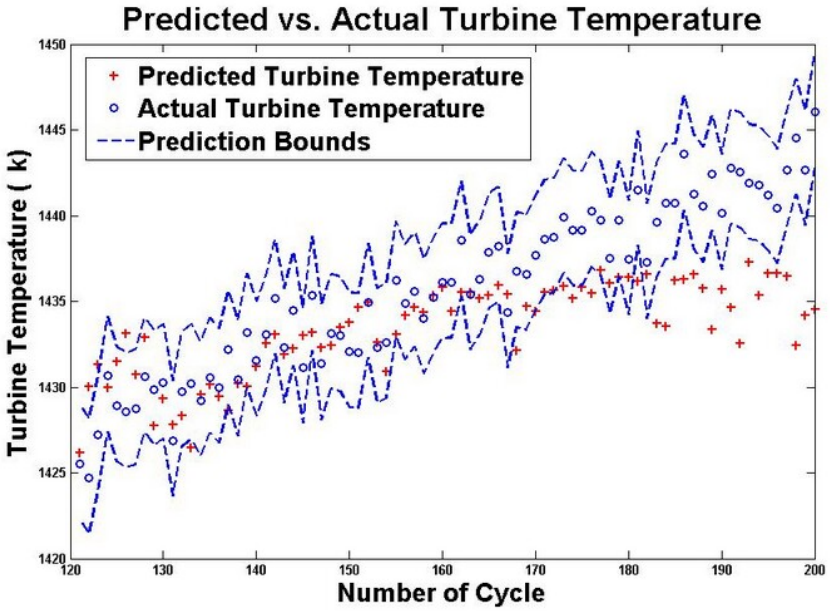

Figure 3.27: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 60% of the available data for $FI = 3\%$.

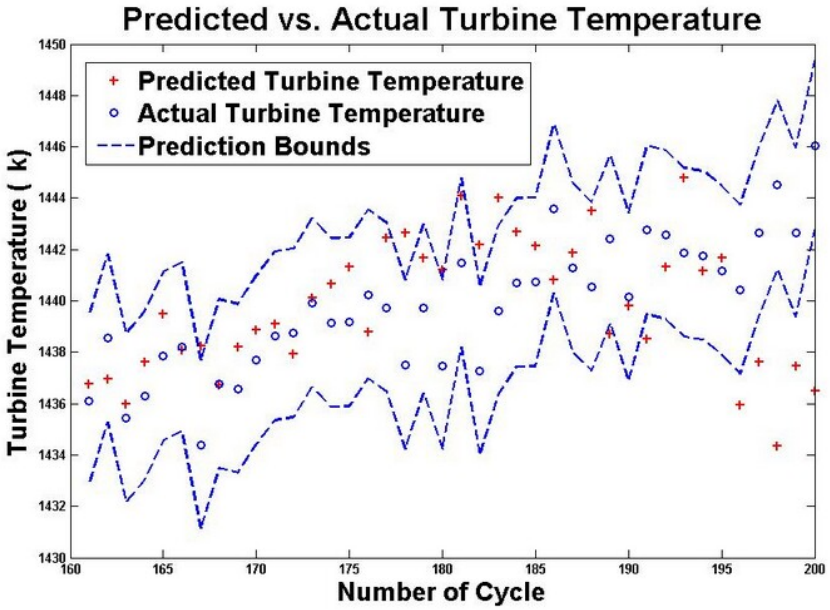

Figure 3.28: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $FI = 3\%$.

Table 3.17: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.5392	3.0461	3.9510
6	**2.8069**	**3.0728**	**3.4816**
7	2.6826	2.8079	3.8707
8	2.1891	3.8489	4.4069
9	3.1645	2.5378	4.0465
10	3.2975	2.9875	4.4370
11	2.8421	3.3518	4.3786
12	2.0212	4.2897	4.7177
13	3.3940	3.4620	4.8327
14	2.3092	3.8981	4.5097
15	0.9744	5.3360	5.3914

Training data are increased to 120 data points. The lowest RMSE in the testing phase is achieved when the number of hidden neurons is 10. The prediction error for different NARX neural network structures are summarized in Table 3.23 where 52.77% of the predicted data points are within the prediction intervals using the network 7-10-1 as shown in Figure 3.30.

Next, 80% of the available data points which is equal to 160 data are used in the training phase and the remaining data points evaluated the network after training. The errors in the evaluation phase are shown in Table 3.24. The mean, standard deviation and RMSE for the network with the structure of 7-7-1 are $3.3750K$, $2.3645K$, and $4.1020K$, respectively. Figure 3.31 shows that 72.22% of the predicted data are within the upper and the lower prediction bounds.

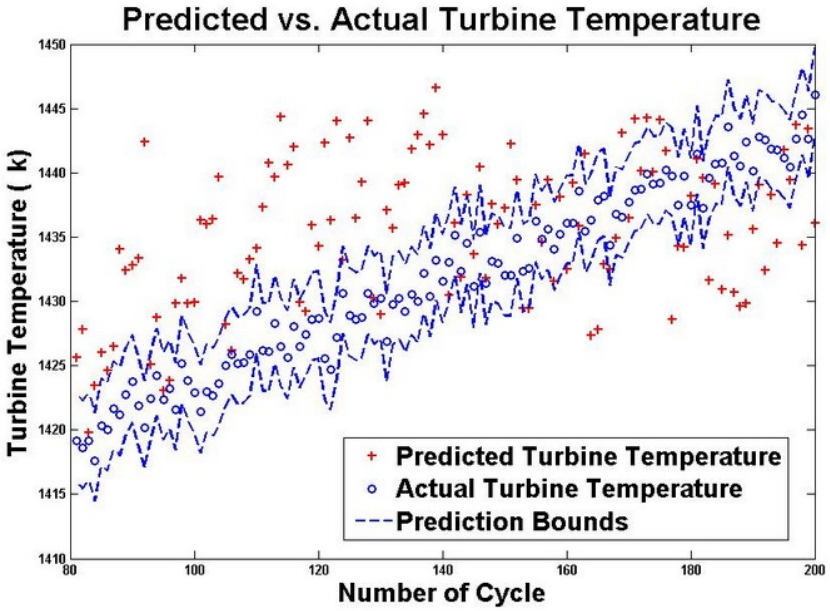

Figure 3.29: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $FI = 3\%$.

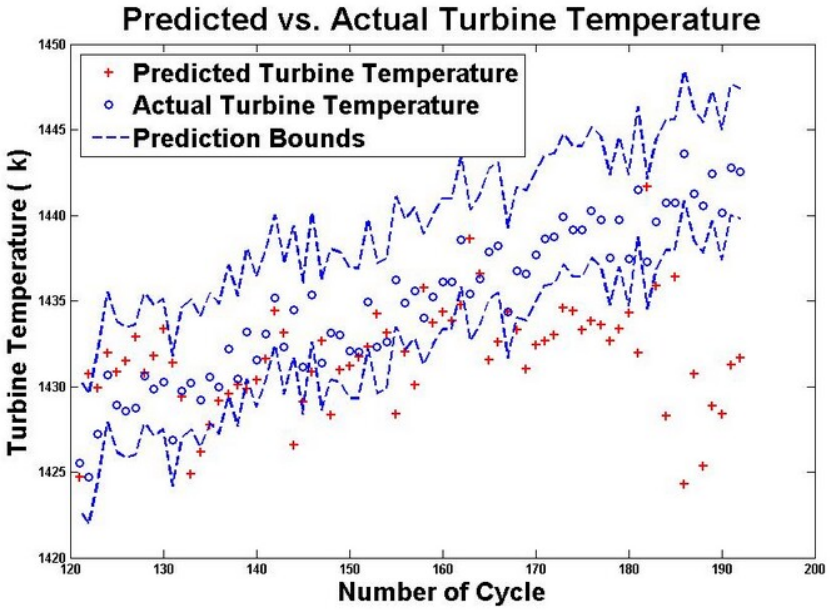

Figure 3.30: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 3\%$.

Table 3.18: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.1949	2.1157	3.8173
6	3.1452	2.0993	3.7668
7	3.0326	2.1551	3.7047
8	2.0988	2.4922	3.2343
9	**2.0149**	**1.8150**	**2.6966**
10	0.6169	3.2538	3.2716
11	1.9793	2.8099	3.4081
12	1.9473	3.0805	3.6117
13	3.2368	2.4540	4.0433
14	2.9768	3.3896	4.4792
15	3.5833	2.5772	4.3950

Table 3.19: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	1.4806	7.6220	7.7332
6	2.1402	7.0301	7.3206
7	2.3948	9.2868	9.5531
8	4.6770	7.9002	9.1524
9	2.3995	7.2184	7.5782
10	**0.6090**	**6.8571**	**6.8556**
11	6.9338	6.6054	9.5575
12	6.4648	8.3889	10.5631
13	7.3220	6.9926	10.1045
14	5.0073	8.0699	9.4685
15	3.4756	9.2582	9.8529

Table 3.20: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.0422	3.1343	5.1030
6	3.5117	3.3120	4.8129
7	**2.2591**	**3.9137**	**4.0757**
8	1.9393	3.8168	4.2599
9	3.7013	3.5748	5.1302
10	3.7432	4.0243	5.4776
11	4.0673	4.0325	5.7098
12	1.7628	5.7050	5.9369
13	3.7274	4.1198	5.5367
14	5.4302	4.0585	6.7640
15	3.8895	6.0622	7.1707

Table 3.21: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.7888	2.5575	3.7623
6	2.7340	2.4544	3.6535
7	**0.0146**	**3.4756**	**3.4319**
8	2.8330	2.5477	3.7888
9	2.9779	2.4867	3.8597
10	1.4539	3.2448	3.5184
11	3.0820	2.7975	4.1387
12	3.2743	2.5403	4.1247
13	3.7790	2.9228	4.7550
14	4.0073	3.5722	5.3386
15	4.7750	2.7100	5.4737

Table 3.22: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.5701	6.6895	10.8546
6	7.8279	7.2783	10.6681
7	4.4410	7.2831	8.5043
8	6.2032	5.6106	8.3485
9	**4.0102**	**7.1050**	**8.1328**
10	4.8240	6.9982	8.4757
11	5.4959	6.9851	8.8651
12	7.4776	6.7458	10.0519
13	7.0705	7.8316	10.5268
14	8.3011	9.2814	12.4231
15	4.9906	11.4128	12.4126

Table 3.23: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	9.5485	4.7101	10.6325
6	8.1959	5.0735	9.6206
7	7.6685	4.9290	9.0975
8	7.7850	4.3816	8.9184
9	6.5033	4.6286	7.9636
10	**4.7436**	**4.2686**	**6.3616**
11	7.5783	4.6664	8.8828
12	7.9640	5.3405	9.5682
13	8.6210	5.1511	10.0243
14	8.0544	6.6133	10.3924
15	9.7400	5.2012	11.0247

Table 3.24: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.7732	2.1194	5.2106
6	3.8800	2.6337	4.6688
7	**3.3750**	**2.3645**	**4.1020**
8	5.2017	2.2642	5.6605
9	4.6815	2.7763	5.4231
10	5.4242	2.1562	5.8260
11	6.0142	2.2808	6.4209
12	5.4897	5.1054	7.4483
13	7.0827	2.7543	7.5856
14	7.4773	2.2253	7.7926
15	7.5507	3.7648	8.4139

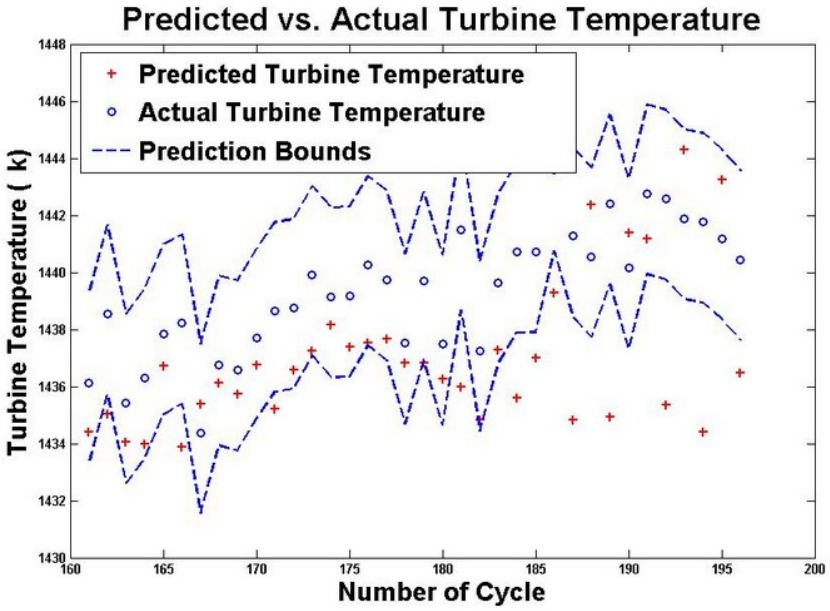

Figure 3.31: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $FI = 3\%$.

3.1.1.3 Summary of the Results

Table 3.25 is the summary of the RMSE for the optimal NARX neural networks found by trial and error in Section 3.1.1.1 in presence of 1% fouling in the compressor for different scenarios, where N_{train} is the number of data used in the training phase and N_{test} is the number of data which were used to test the trained network. Finally, N_F is the number of flights ahead which the networks are used to predict their turbine temperature. From Table 3.25, it can be seen that as the training data increases, the prediction error decreases. The RMSE decreases 37.6% when the training data increases from 80 to 160 data points for 5 flights ahead. Also, there are 60.83% of the predicted data within the upper and the lower prediction bounds when the network is trained by using 80 data points. However, this value increases to 77.5% when the network is trained with 160 data points. The prediction errors also increase when the turbine temperatures are predicted for more flights ahead. The RMSE increases by 46.76% when the network predicts 12 flights ahead in comparison to the prediction of 2 flights ahead turbine temperatures.

Table 3.25: Summary of the prediction errors for each scenario in presence of $FI = 1\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-6-1	3.4998
120	80	2	7-5-1	2.6993
160	40	2	7-6-1	1.6919
80	120	5	7-7-1	3.8167
120	80	5	7-8-1	3.1950
160	40	5	7-6-1	2.3814
80	120	8	7-8-1	4.1984
120	80	8	7-8-1	3.3688
160	40	8	7-6-1	2.7554
80	120	12	7-9-1	5.2964
120	80	12	7-8-1	3.8929
160	40	12	7-7-1	3.1779

The summary of the prediction error results for the optimal networks which are found for each scenario in presence of 3% compressor fouling is also presented in Table 3.26 where the prediction error increases by 51.38% when the network predicts 12 flights ahead turbine temperature in comparison to 2 flights ahead. The RMSE decreases when the network is trained by using more data points. The RMSE decreases 49.5% when the NARX neural network is trained by using 160 data points rather than 80 data points in the 12 flights ahead prediction.

Table 3.26: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-8-1	4.4192
120	80	2	7-9-1	3.1743
160	40	2	7-8-1	2.1079
80	120	5	7-9-1	5.5322
120	80	5	7-6-1	3.4816
160	40	5	7-9-1	2.6966
80	120	8	7-10-1	6.8556
120	80	8	7-7-1	4.0757
160	40	8	7-7-1	3.4319
80	120	12	7-9-1	8.1328
120	80	12	7-10-1	6.3616
160	40	12	7-7-1	4.1020

3.1.2 Turbine Erosion

As mentioned in Section 2.6.4, turbine erosion is one of the main causes of degradation in turbine section in aero engine applications. Erosion decreases the turbine efficiency and increases its mass flow rate due to the loss of materials in the flow path. In this section, it is assumed that the turbine of the jet engine goes under 2 different percentages of erosion, namely 1% and 3% during 200 flights. These degradations are simulated using equations (2.6.3) and (2.6.4) in our jet engine model described in Section 2.5. These degraded data are collected and used to train the NARX networks under different structures. The trained networks are fixed to be used in the evaluation process to check the reliability of these networks in predicting the turbine temperature in presence of turbine erosion.

3.1.2.1 **EI** = 1%

The engine goes through 1% erosion in 200 flights which implies that the efficiency decreases by 1% and the turbine mass flow rate increases by 0.5% where 80 data points are used to train different NARX neural network structures to predict 2 flights ahead turbine temperature. These networks are then tested using 120 data points. The prediction error for these networks are tabulated in Table 3.27. The prediction and actual values as well as prediction bounds for the network with the structure of 7-6-1 are shown in Figure 3.32 where 73.33% of the predicted data are within the prediction bounds. The errors which are the absolute values of the differences between the actual and predicted data points are depicted in Figure 3.33.

Training data are increased to 120 data points and different NARX neural network structures are trained. The trained networks are then tested with 60 unseen data points. The results of errors in the testing phase are summarized in Table 3.28. The network with 8 hidden neurons has the lowest RMSE. 97.5% of predicted data are

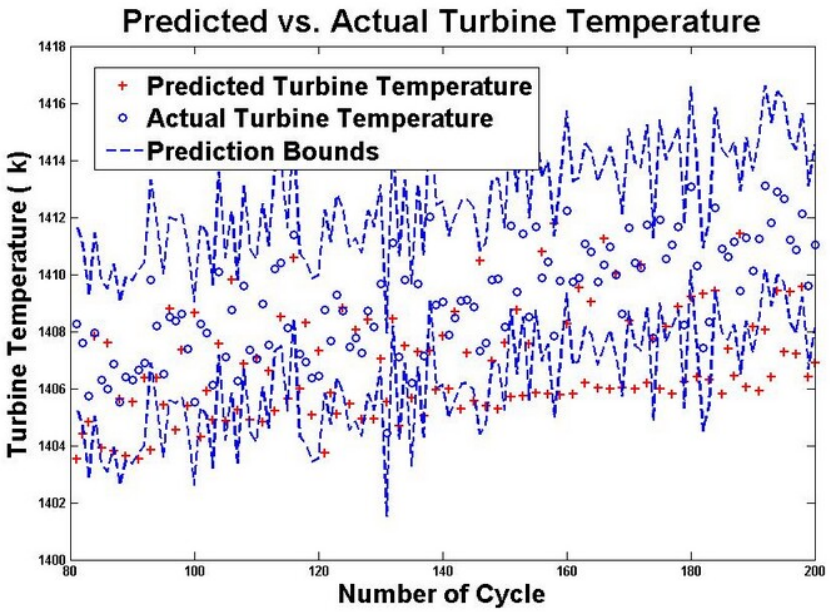

Figure 3.32: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 40% of the available data for $EI = 1\%$.

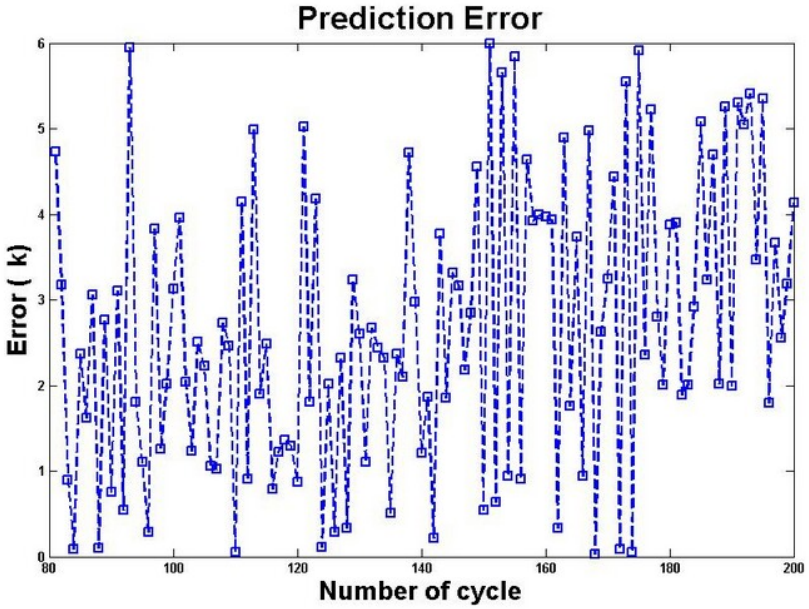

Figure 3.33: The prediction errors for the 2 step ahead turbine temperature when $EI = 1\%$ using NARX 7-6-1 trained with 40% of the available data.

Table 3.27: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.8623	1.9413	3.4540
6	**2.2700**	**2.1630**	**3.1293**
7	2.8107	1.7435	3.3037
8	3.1144	2.5174	3.9980
9	3.0551	1.9717	3.6316
10	3.3536	2.0735	3.9383
11	3.9016	1.8903	4.3320
12	4.2378	1.9584	4.6650
13	3.5411	3.3325	4.8531
14	3.9263	2.7340	4.7779
15	4.2506	3.0413	5.2192

within the prediction intervals in this case as shown in Figure 3.34.

In the case when the training data are increased from 120 to 160 data points, the RMSE is $1.5919K$ when the NARX neural network structure is 7-6-1. Comparing Tables 3.27 and 3.29, the RMSE decreases 50.8% by increasing the training data from 80 to 160 data points. Actual and predicted values along with the prediction bounds are depicted in Figure 3.35 where 100% of the predicted data in the testing phase are within the prediction bounds.

Next the 5 flights ahead turbine temperature is now predicted using different numbers of the training data for various NARX neural network architectures. The results of the prediction error, standard deviation and RMSE are tabulated in Tables 3.30-3.32 for the networks trained by using 40%, 60%, and 80% of the entire available data. In the case when the network is trained by using 40% data points, 64.1% of the predicted data are within the prediction bounds as shown in Figure 3.36. This value increases to 93.75% for the case when the network is trained by using 60% of the available data points, and finally 95% for the network trained by using 80% of

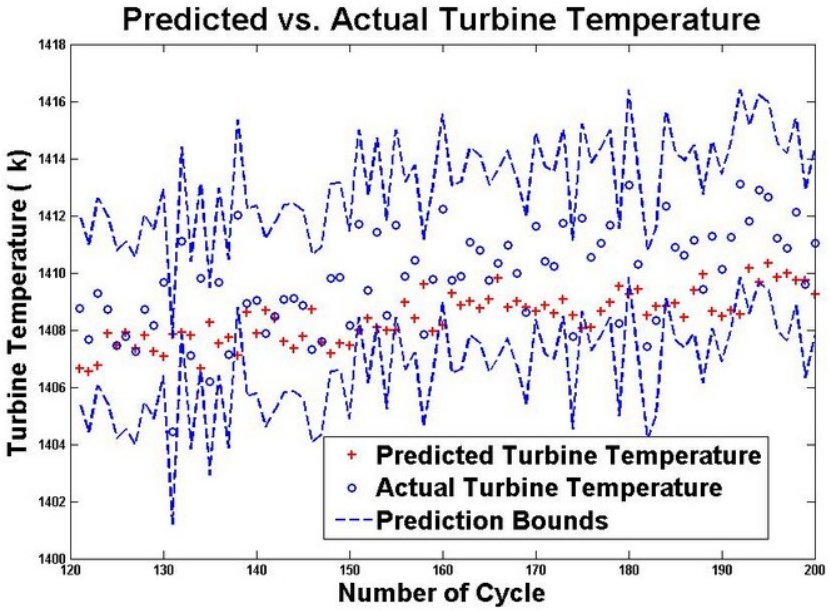

Figure 3.34: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $EI = 1\%$.

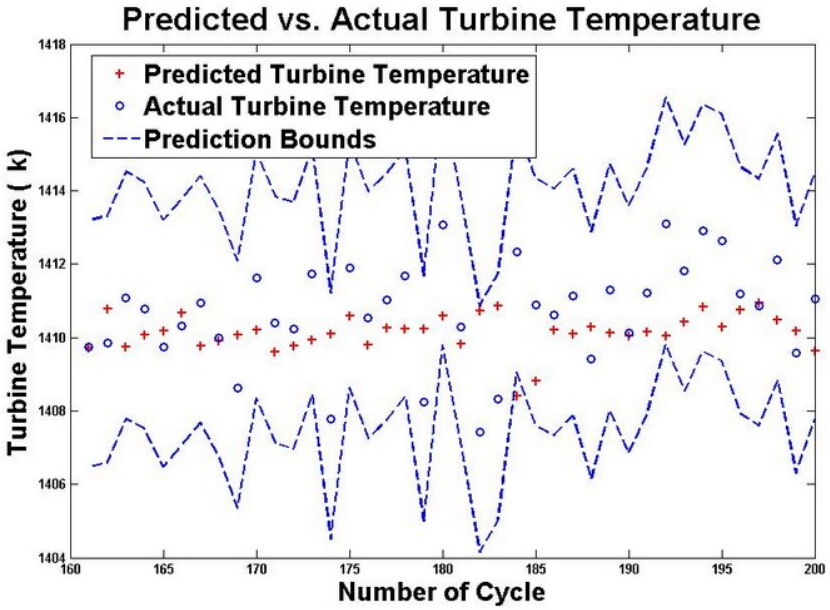

Figure 3.35: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 80% of the available data for $EI = 1\%$.

Table 3.28: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	0.5030	2.5982	2.6304
6	1.6985	1.8051	2.4704
7	1.3883	1.8247	2.2837
8	**1.3862**	**1.6132**	**2.1193**
9	1.8911	1.6051	2.4740
10	1.7355	1.6814	2.4091
11	1.8775	1.8359	2.6180
12	2.0234	2.0686	2.8844
13	2.8733	1.9560	3.4690
14	0.6462	3.3333	3.3748
15	3.3266	2.1205	3.9378

the available data points as depicted in Figure 3.38.

Next the 8 flight ahead turbine temperature is also predicted in presence of 1% turbine erosion for maintenance actions. Various NARX neural networks are trained with 80 data points and each of them is evaluated with 120 data points. The prediction errors are tabulated in Table 3.33. It can be seen that the network with 8 hidden neurons has the lowest RMSE. The data used in the testing phase as well as predicted values are depicted in Figure 3.39 where 57.5% of the predicted data points are between the prediction bounds.

The training data increased from 40% of the available data points to 60%. Table 3.34 summarizes the prediction errors when different neural network structures are trained. The number of hidden neurons increases from 5 to 15. The difference between the actual and predicted values for each testing data points for the NARX neural network structure of 7-5-1 are depicted in Figure 3.40 where 13.75% of the predicted data points are outside the prediction bounds as shown in Figure 3.41.

When the NARX neural networks are trained by using 80% of the entire data, the

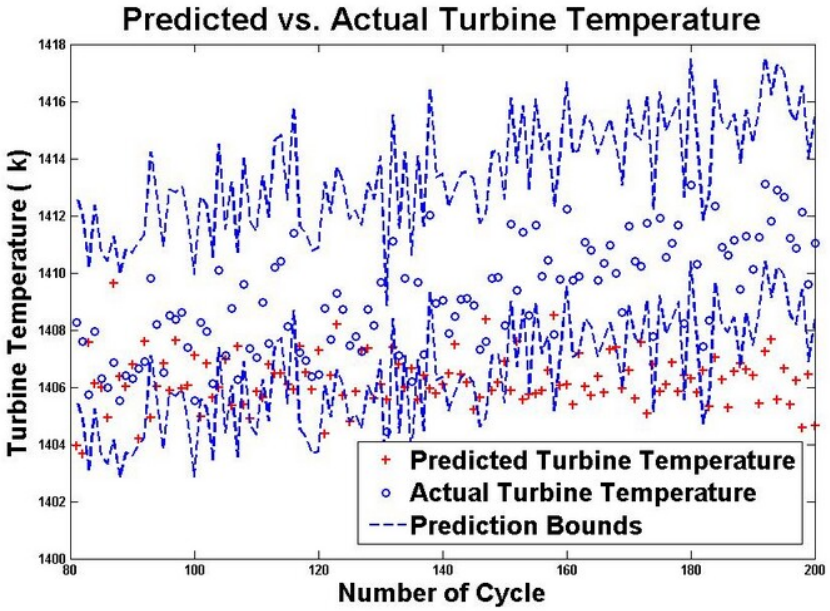

Figure 3.36: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $EI = 1\%$.

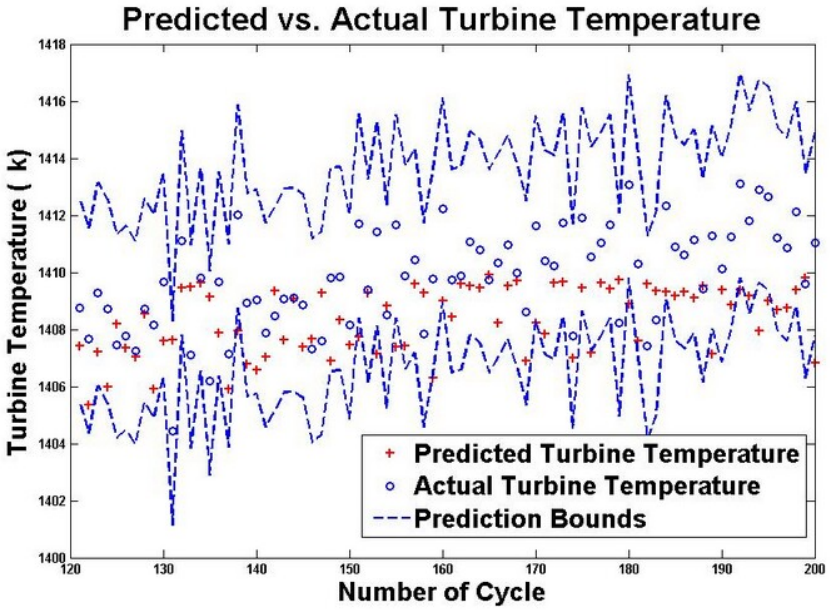

Figure 3.37: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $EI = 1\%$.

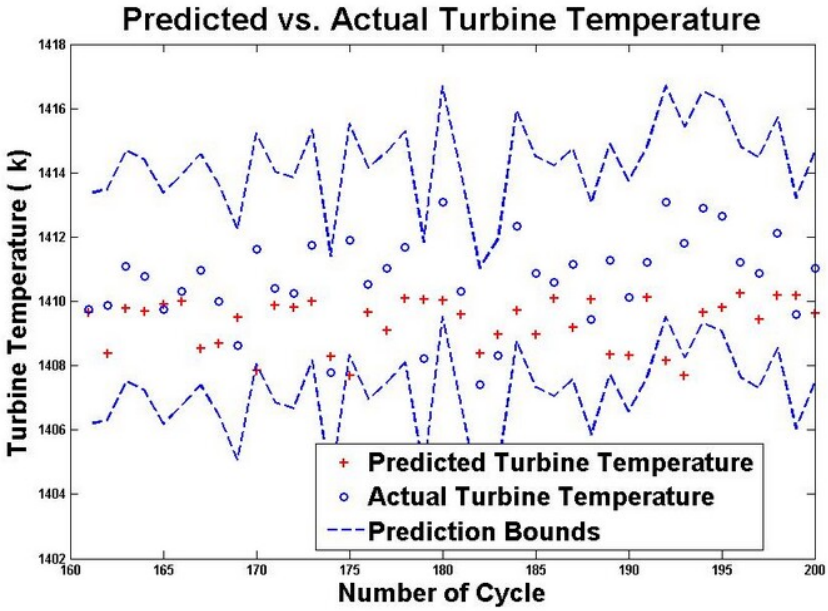

Figure 3.38: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 80% of the available data for $EI = 1\%$.

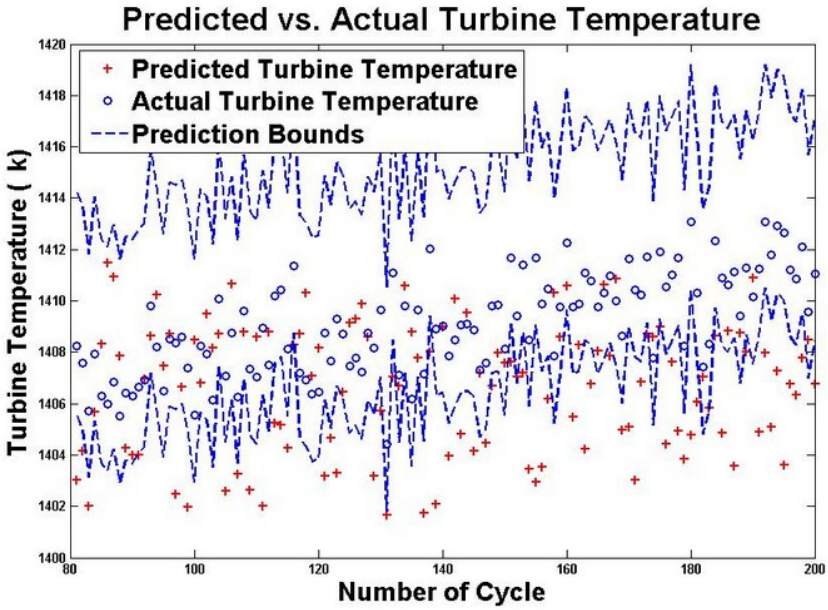

Figure 3.39: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 40% of the available data for $EI = 1\%$.

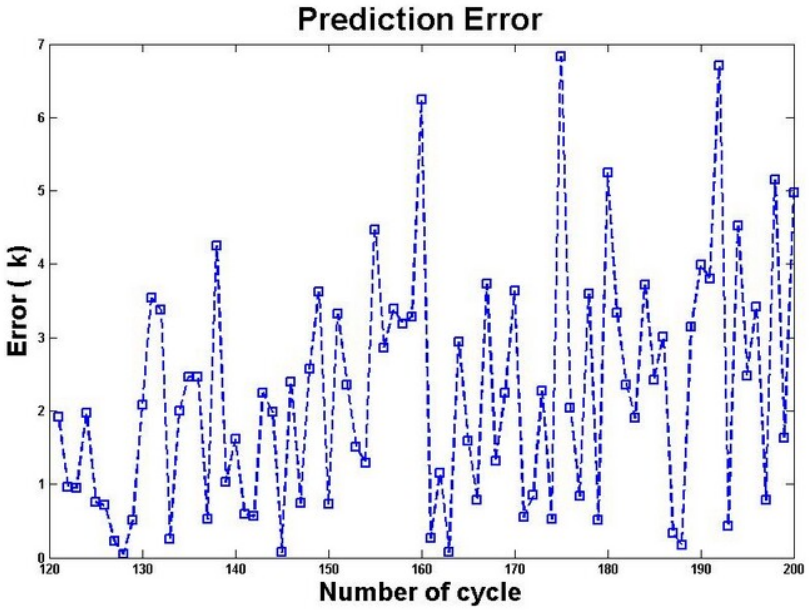

Figure 3.40: The prediction error for the 2 step ahead turbine temperature when $EI = 1\%$ using NARX 7-5-1 trained with 60% of the available data.

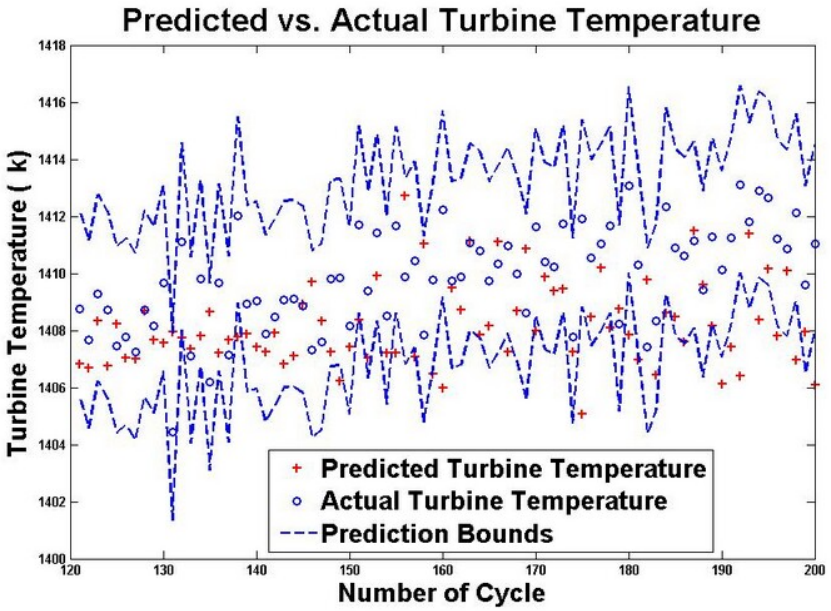

Figure 3.41: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-5-1 trained with 60% of the available data for $EI = 1\%$.

Table 3.29: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	1.3475	1.9249	2.3299
6	**0.5570**	**1.5102**	**1.5919**
7	1.4735	1.5154	2.1001
8	1.0584	1.6019	1.9032
9	1.8197	1.4606	2.3219
10	1.2731	2.2806	2.5869
11	2.0304	1.6468	2.6012
12	2.3349	1.8327	2.9541
13	2.1228	2.0314	2.9206
14	1.1273	2.9998	3.1693
15	2.9755	1.8015	3.4667

prediction error, standard deviation and RMSE for the network with the structure of 7-7-1 are $1.8553K$, $1.8776K$, and $2.6229K$, respectively as shown in Table 3.35. Figure 3.42 shows that 92.5% of the data points predicted by this network are between the prediction bounds.

The applicability of the NARX neural network to predict 12 flights ahead turbine temperature in presence of 1% erosion is investigated. Tables 3.36-3.38 summarize the results of prediction errors when different network structures are trained by using 40%, 60%, and 80% of the entire data sets. Actual versus predicted values as well as prediction bounds are depicted in Figures 3.43-3.45, respectively where 48.33% of the predicted values are within the prediction bounds when 80 data used for the training phase. This value increases to 77.7% for the network trained by using 60% of the available data in the training phase, and finally 83.33% of the predicted data are within the upper and the lower bounds when 80% of the entire data used in the training step.

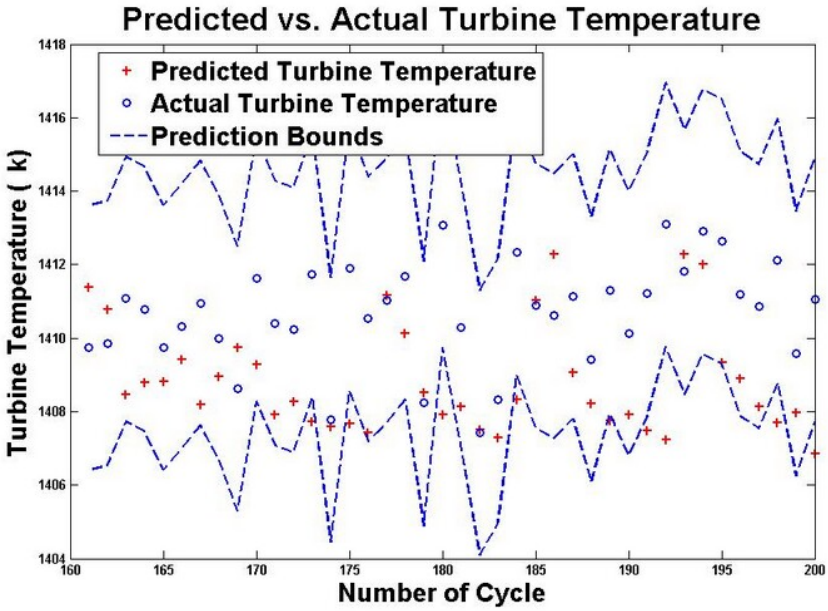

Figure 3.42: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $EI = 1\%$.

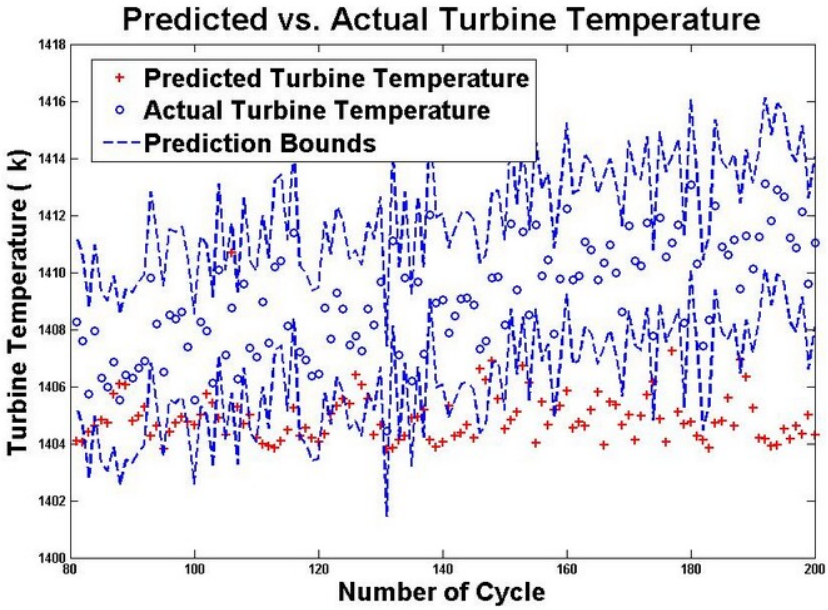

Figure 3.43: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $EI = 1\%$.

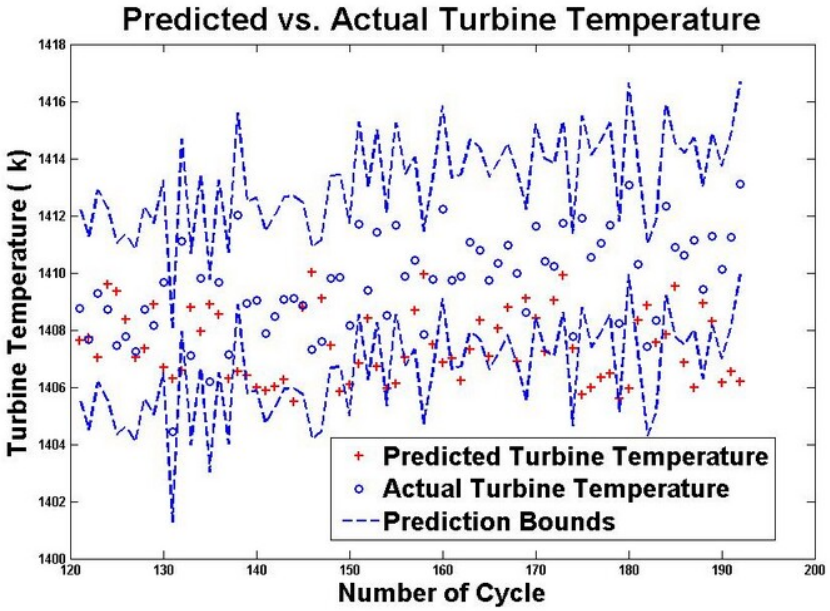

Figure 3.44: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $EI = 1\%$.

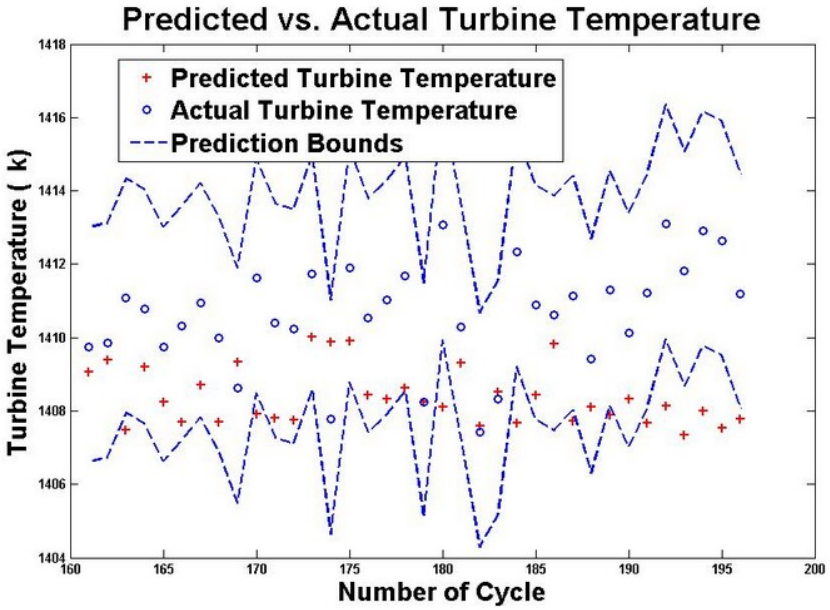

Figure 3.45: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $EI = 1\%$.

Table 3.30: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.5988	2.6025	5.2788
6	4.3960	2.3230	4.9675
7	3.8219	2.0373	4.3270
8	3.3500	2.5087	4.1790
9	**2.8606**	**2.2435**	**3.6296**
10	3.7378	2.0293	4.2491
11	3.4051	2.3471	4.1301
12	3.4195	2.5700	4.2711
13	3.5572	2.6709	4.4416
14	4.1008	2.2961	4.6952
15	3.7128	3.0967	4.8264

Table 3.31: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.7324	1.8346	3.2848
6	2.1921	1.8207	2.8424
7	2.3129	1.7455	2.8911
8	**1.4614**	**1.7887**	**2.3012**
9	2.0728	1.9379	2.8293
10	0.5133	3.1026	3.1256
11	3.0783	1.8863	3.6041
12	3.3039	2.0712	3.8926
13	1.9220	3.6856	4.1362
14	3.4871	2.4862	4.2736
15	3.0048	3.0236	4.2493

Table 3.32: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	1.9924	1.5999	2.5427
6	**1.3610**	**1.5532**	**2.0505**
7	1.0403	1.8869	2.1339
8	1.5454	1.6056	2.2140
9	1.8519	1.6287	2.4528
10	1.6609	1.8300	2.4543
11	1.9521	1.7254	2.5910
12	2.2251	2.0162	2.9858
13	2.2500	2.0848	3.0496
14	1.2672	2.9349	3.1629
15	2.8312	2.0684	3.4911

Table 3.33: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.6912	2.0861	5.1306
6	4.1602	2.2064	4.7048
7	3.4450	1.9966	3.9775
8	**2.4470**	**3.0824**	**3.9255**
9	2.5230	3.0212	3.9265
10	3.8299	2.6837	4.6702
11	3.7724	2.8155	4.7002
12	3.9295	2.7277	4.7769
13	4.5168	1.9612	4.9209
14	4.3267	2.5882	5.0362
15	4.6410	3.5574	5.8385

Table 3.34: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	**1.6487**	**2.2204**	**2.7544**
6	2.3507	1.8535	2.9863
7	2.8059	2.1036	3.4990
8	3.0556	1.7725	3.5269
9	3.4438	1.8377	3.8981
10	2.7760	2.9345	4.0262
11	2.4178	3.7004	4.4009
12	1.7547	4.2697	4.5915
13	4.3013	2.3044	4.8729
14	0.5725	5.2885	5.2864
15	1.0745	5.3360	5.4103

Table 3.35: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.9267	1.5197	3.2890
6	2.1836	2.3608	3.1941
7	**1.8553**	**1.8776**	**2.6229**
8	1.4631	2.6298	2.9806
9	2.2617	1.9330	2.9594
10	2.7368	1.7742	3.2495
11	3.0637	1.4892	3.3983
12	3.3935	1.6826	3.7784
13	3.5519	1.5845	3.8812
14	3.9267	2.0163	4.4026
15	3.9536	2.4690	4.6449

Table 3.36: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.3057	3.5868	6.3960
6	4.2327	3.9129	5.7531
7	4.7890	2.3851	5.3456
8	4.2782	2.6544	5.0289
9	4.4607	2.4686	5.0932
10	**4.1774**	**2.1854**	**4.7103**
11	4.3861	2.3107	4.9531
12	3.0718	4.1297	5.1331
13	4.7999	2.7709	5.5365
14	5.2038	2.6367	5.8287
15	2.7625	5.7629	6.3691

Table 3.37: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.7320	2.0376	4.2453
6	3.4639	1.9900	3.9880
7	2.8529	1.6928	3.3113
8	**2.1995**	**2.3179**	**3.1837**
9	2.5916	2.7420	3.7591
10	2.0664	3.1369	3.7381
11	2.4978	3.2655	4.0932
12	3.4248	2.7661	4.3902
13	3.9087	2.6898	4.7342
14	4.1744	4.1809	5.8875
15	5.4344	2.8158	6.1115

Table 3.38: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.7933	2.0983	3.4761
6	2.3628	2.1129	3.1500
7	**2.2907**	**1.7558**	**2.8713**
8	0.0855	3.1904	3.1470
9	3.1614	1.9377	3.6939
10	3.4053	1.6877	3.7901
11	3.3826	2.5650	4.2236
12	4.0232	1.8432	4.4146
13	2.4607	3.8335	4.5103
14	4.0813	1.9710	4.5204
15	3.9465	2.7737	4.8015

3.1.2.2 EI = 3%

In this case the engine goes through 3% erosion in 200 flights which implies that the efficiency decreases by 3% and the turbine mass flow rate increases by 1.5%. Data are generated from our Simulink model which was described in Section 2.6.4 and based on equations (2.6.3) and (2.6.4). As was done in the previous sections, 80 data points are used to train different NARX neural network structures. The weights and biases in these networks remain fixed and 120 unseen data points are given to the networks to predict 2 flights ahead turbine temperature. The prediction error, standard deviation and RMSE are presented in Table 3.39 when the number of hidden neurons changes from 5 to 15, where d_u and d_y both set to 3. Based on Table 3.39, the network with 11 hidden neuron has the smallest RMSE equal to $3.8966K$. Figure 3.46 shows the actual and predicted values pointwise which shows that 56.66% of the predicted data are within the prediction intervals.

Table 3.39: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.5079	3.4763	6.5055
6	4.6412	3.1878	5.6230
7	4.0556	3.5088	5.3532
8	2.8480	3.6282	4.6006
9	3.4491	3.1332	4.6510
10	2.9257	2.8709	4.0906
11	**1.2225**	**3.7154**	**3.8966**
12	3.0691	3.2702	4.4749
13	1.8226	4.3508	4.7003
14	0.1346	5.1627	5.1429
15	4.2339	3.5664	5.5262

When more data are used in the training phase, the RMSE decreases as shown in Table 3.40 where 60% of the entire data are used to train the networks and 40% to

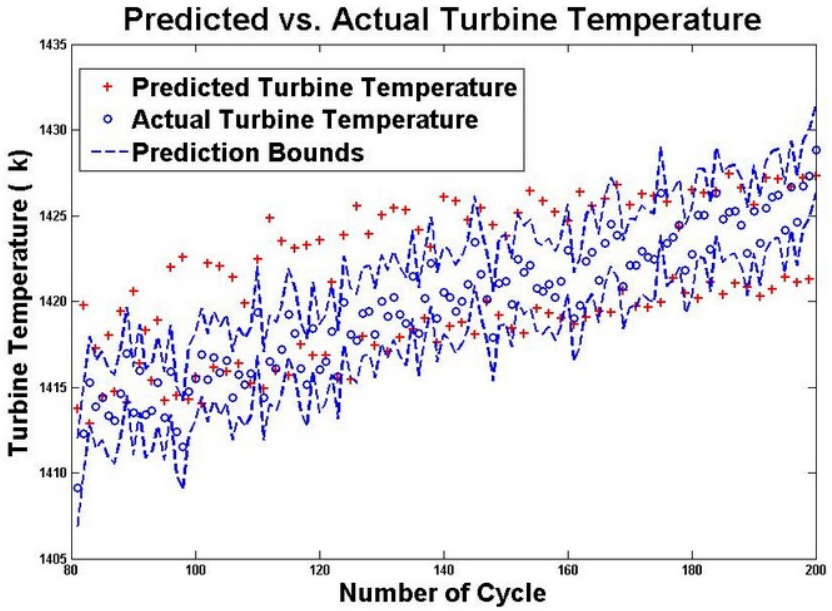

Figure 3.46: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 40% of the available data for $EI = 3\%$.

evaluate the networks in turbine temperature prediction. The actual and predicted values for the NAXR neural network structure of 7-8-1 are depicted in Figure 3.47 where 86.66% of the data are between the upper and the lower prediction bounds. Figure 3.48 shows the absolute difference between the actual and predicted value for each data point.

If the training data increase from 120 data points to 160 data points, the RMSE of the optimal NARX neural network is $2.4922K$ for 2 flights ahead prediction. The results of the prediction error for different structures are tabulated in Table 3.41. From Figure 3.49, it is clear that only 7.5% of the prediction data are outside the prediction intervals.

Various NARX neural networks are trained and tested to predict 5 flights ahead turbine temperature. As done for the previous cases, different number of training and testing data are used to find the optimal NARX neural network structure. The

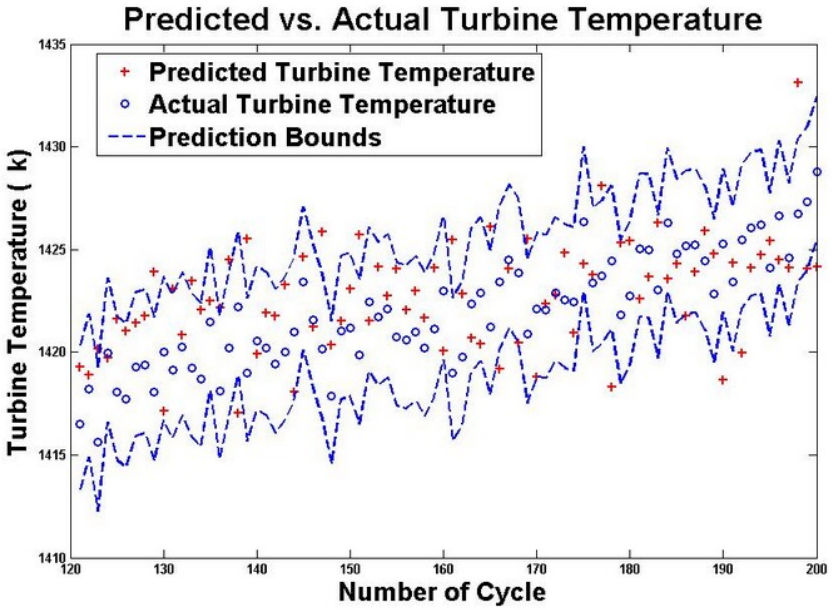

Figure 3.47: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $EI = 3\%$.

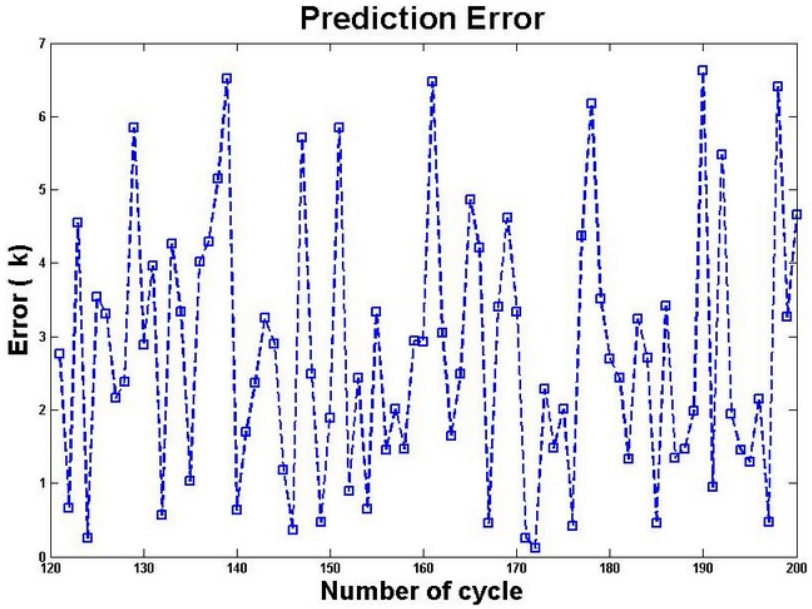

Figure 3.48: The prediction error for the 2 step ahead turbine temperature when $EI = 3\%$ using NARX 7-8-1 trained with 60% of the available data.

Table 3.40: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.4051	2.3483	4.9850
6	0.6855	4.2737	4.3019
7	3.7471	1.9493	4.2182
8	**0.7661**	**3.1798**	**3.2514**
9	3.5696	2.1840	4.1776
10	3.7060	3.2697	4.9287
11	4.5460	2.3660	5.1180
12	4.6960	2.1244	5.1487
13	4.6764	2.4078	5.2530
14	4.6573	3.4525	5.7845
15	4.9690	3.0656	5.8285

results of prediction error when 80 data points are used in the training phase and 120 points are used in the testing phase are shown in Table 3.42. The predicted data using the network 7-9-1 versus the actual values along with their prediction intervals are depicted in Figure 3.50 where only 59.1% of the predicted data points are within the upper and the lower bounds. When the number of training data increases to 120 points, the NARX neural network with 6 hidden neurons has the lowest RMSE as presented in Table 3.43 where 66.25% of the predicted data are within the prediction intervals for this network as depicted in Figure 3.51. The RMSE decreases to $2.7411K$ in presence of 80% training data as shown in Table 3.44. The 40 predicted data are shown pointwise in Figure 3.52. It can be seen that only 20% of the data points are outside the prediction bounds. The absolute difference between the actual and predicted data points are shown in Figure 3.53.

Next the 8 flights ahead turbine output temperature is now predicted using different NARX neural network structures with various numbers of training and testing data. In the first case the NARX neural networks are trained by using 40% of the

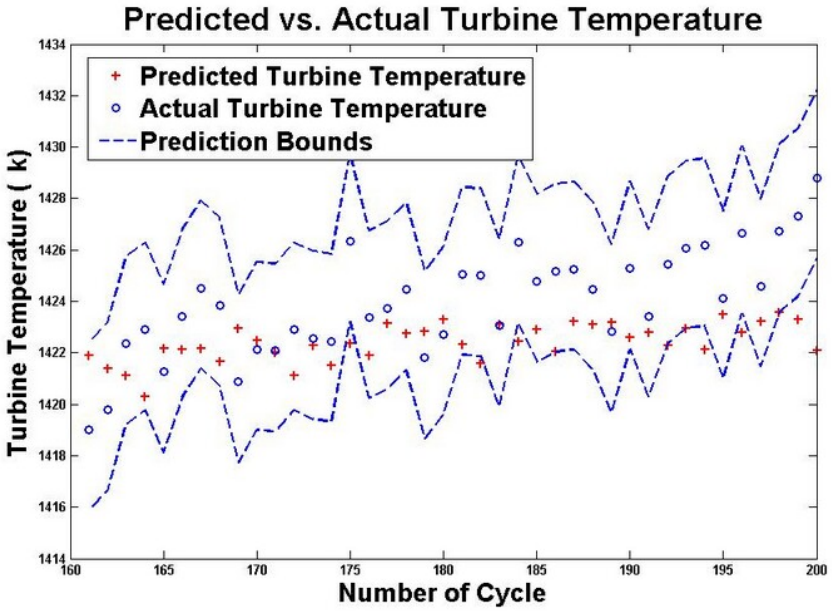

Figure 3.49: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 80% of the available data for $EI = 3\%$.

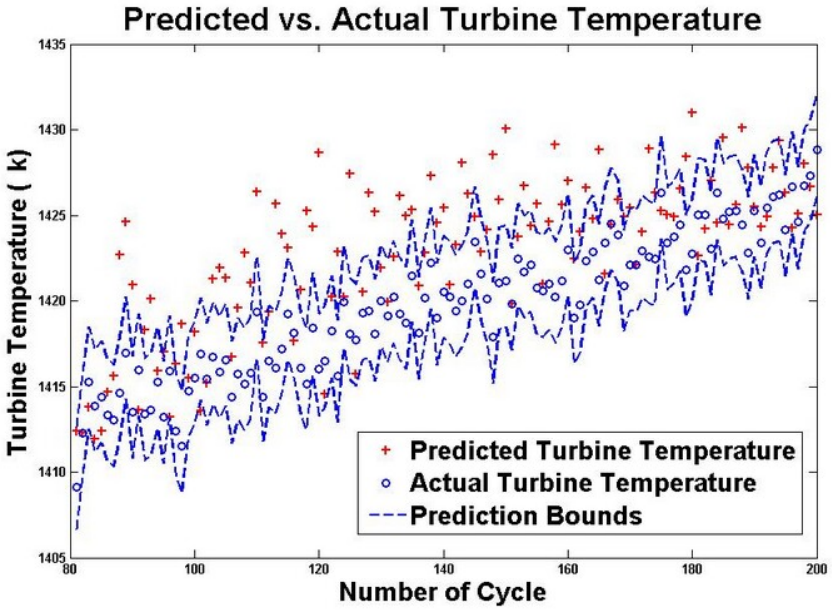

Figure 3.50: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $EI = 3\%$.

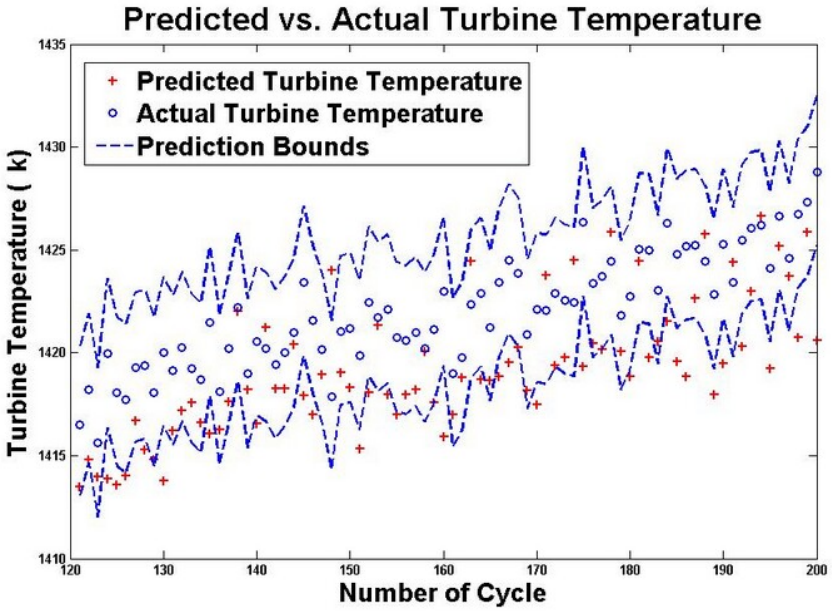

Figure 3.51: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 60% of the available data for $EI = 3\%$.

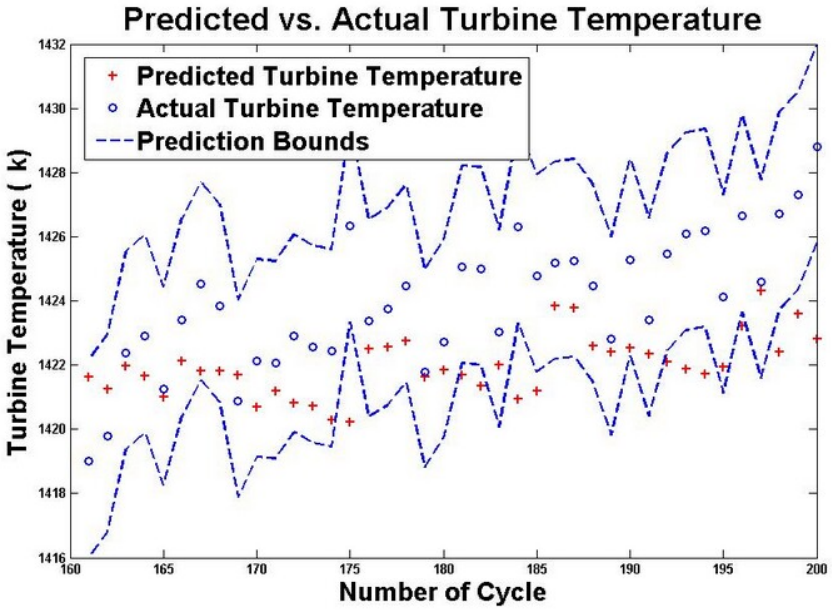

Figure 3.52: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $EI = 3\%$.

Table 3.41: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.8643	2.0853	4.3787
6	2.0448	3.5923	4.0943
7	1.7942	1.8558	2.5645
8	**1.5597**	**1.9685**	**2.4922**
9	2.1645	1.9895	2.9231
10	2.8994	1.9287	3.4689
11	2.6922	2.3187	3.5341
12	1.1806	3.4754	3.6291
13	3.0490	2.0765	3.6743
14	3.4783	2.0686	4.0337
15	3.5430	2.0709	4.0908

total available data points (equal to 80), and 120 remaining data points are used in the evaluation process. The results of the prediction error for these networks are summarized in Table 3.45. As depicted pointwise in Figure 3.54, only 38.33% of the predicted points are within the prediction bounds.

The results of the prediction error in the case when the neural networks are trained by using 120 points are shown in Table 3.46. The lowest RMSE is achieved when the network has 7 inputs, 8 hidden neurons and 1 output where 45.2% of the predicted data points are between the lower and the upper prediction intervals as depicted in Figure 3.55.

When the training data increase to 160 data points, the RMSE is $4.3814K$ with the network with 7 hidden neurons as presented in Table 3.47. The actual and predicted data points are depicted pointwise in Figure 3.56 where 87.5% of the predicted points are within the prediction bounds.

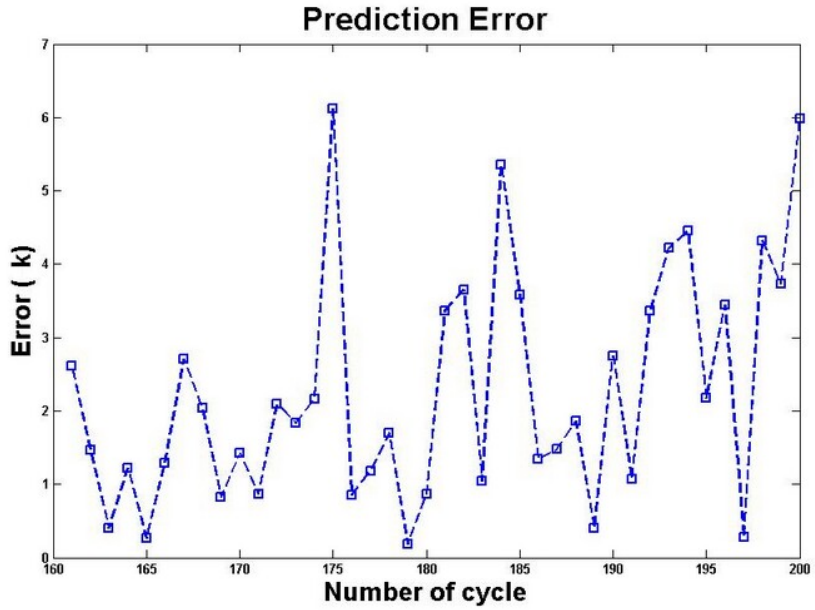

Figure 3.53: The prediction error for the 2 step ahead turbine temperature when $EI = 3\%$ using NARX 7-7-1 trained with 80% of the available data.

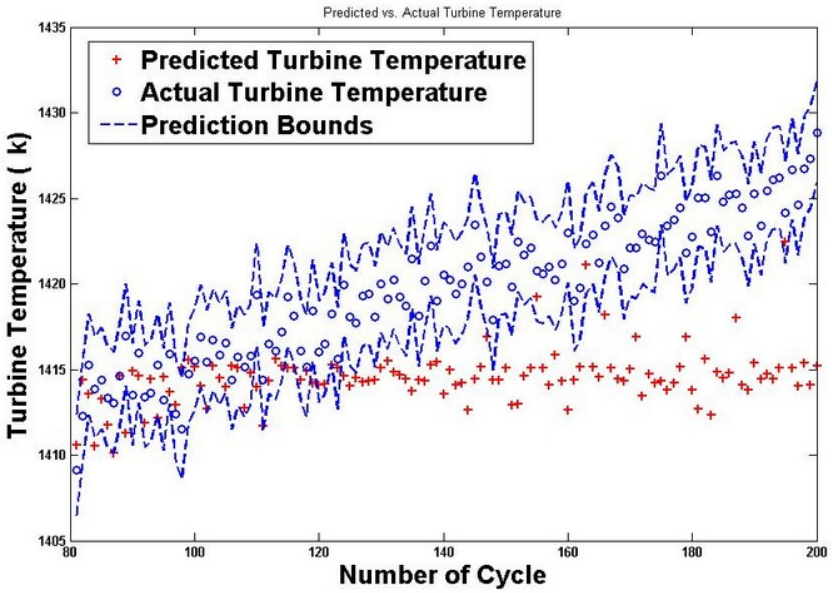

Figure 3.54: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $EI = 3\%$.

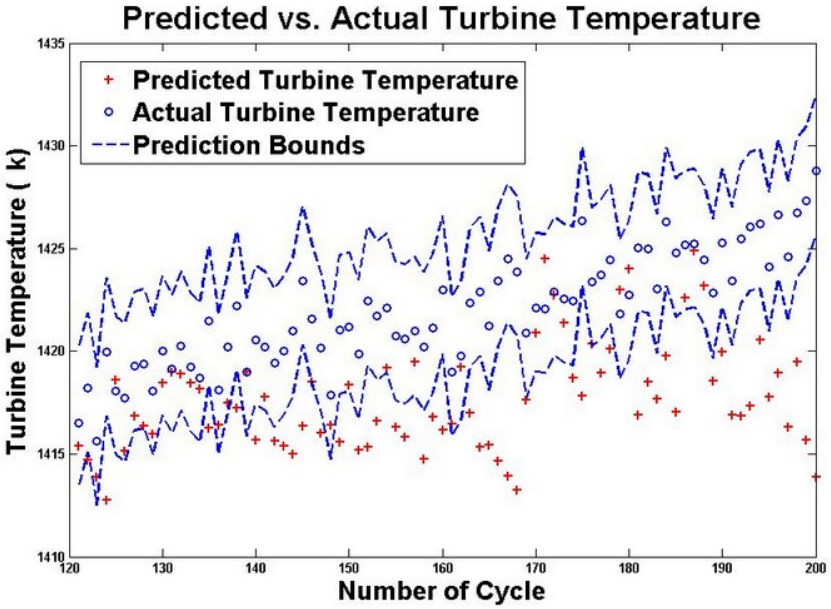

Figure 3.55: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 60% of the available data for $EI = 3\%$.

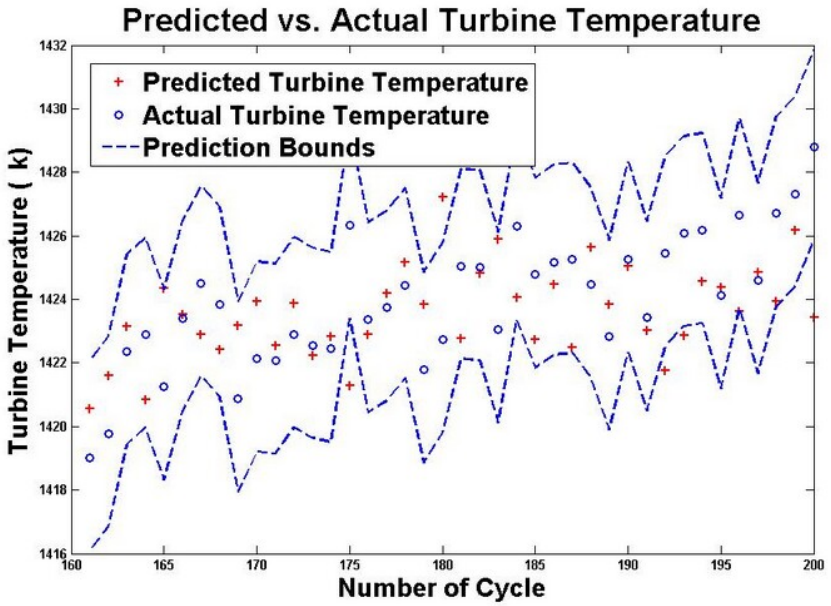

Figure 3.56: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $EI = 3\%$.

Table 3.42: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.6393	4.1490	6.2124
6	5.0700	3.7797	6.3144
7	4.9200	3.6521	6.1182
8	3.8595	3.4692	5.1798
9	**3.1685**	**3.3852**	**4.6264**
10	4.9004	3.3789	5.9444
11	4.9753	4.4286	6.6485
12	5.5353	4.0317	6.8380
13	2.1620	7.1622	7.4528
14	6.9197	3.8536	7.9125
15	5.7768	5.5969	8.0272

Table 3.43: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.6517	2.7593	5.3997
6	**2.7987**	**3.4564**	**3.7137**
7	2.5396	2.7658	3.7421
8	2.7939	2.6335	3.8281
9	2.2628	3.7053	4.3218
10	4.1963	2.2525	4.7560
11	4.7191	2.5128	5.3391
12	5.2444	2.8377	5.9545
13	5.6038	2.6784	6.2038
14	5.3212	4.4639	6.9277
15	7.0162	3.3228	7.7544

Table 3.44: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	2.3539	2.1542	3.1726
6	2.0199	2.0381	2.8513
7	**2.0044**	**1.8936**	**2.7411**
8	2.1379	1.8844	2.8342
9	2.4710	2.0218	3.1767
10	2.5934	2.2455	3.4121
11	2.9812	2.5353	3.8929
12	3.2062	2.4969	4.0445
13	3.8146	2.3065	4.4428
14	3.9130	2.6768	4.7221
15	4.3169	2.3404	4.8965

Table 3.45: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.4934	4.1325	9.4378
6	8.2054	3.7542	9.0170
7	7.5029	4.4063	8.6917
8	7.1344	4.4559	8.4018
9	6.2279	3.7142	7.2434
10	**5.2000**	**3.8503**	**6.4607**
11	6.6509	4.1424	7.8263
12	5.9541	6.0553	8.4742
13	8.3803	5.0252	9.7608
14	8.5948	5.3559	10.1152
15	8.9073	5.1604	10.2833

Table 3.46: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.7430	2.6444	6.3156
6	5.1091	2.6671	5.7556
7	**4.3238**	**3.2193**	**5.3786**
8	5.3898	2.9766	6.1481
9	5.8761	2.6279	6.4302
10	5.9924	2.7578	6.5893
11	6.0463	3.0360	6.7572
12	6.4100	3.3204	7.2094
13	6.4213	3.2955	7.2081
14	6.5856	3.1116	7.2753
15	6.5101	3.5443	7.4018

Table 3.47: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.6737	2.4102	5.2447
6	3.2405	2.9587	4.3630
7	**3.3743**	**2.8303**	**4.3814**
8	3.6696	2.3619	4.3480
9	4.0057	2.3861	4.6472
10	4.8206	2.5577	5.4421
11	4.8994	2.5980	5.5303
12	3.6908	4.3411	5.6565
13	4.3453	3.4211	5.5038
14	3.3675	5.2238	6.1600
15	5.1067	3.3798	6.1005

3.1.2.3 Summary of the Results

Tables 3.48 and 3.49 show the summary of the RMSE for the optimal NARX neural networks found by trial and error in Sections 3.1.2.1 and 3.1.2.2 in presence of 1% and 3% turbine erosion for different scenarios, where N_{train} is the number of data used in the training phase and N_{test} is the number of data which were used to test the trained network, and N_F is the number of flights ahead which the networks are used to predict the turbine temperature.

Table 3.48: Summary of the prediction errors for each scenario in presence of $EI = 1\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-6-1	3.1293
120	80	2	7-8-1	2.1193
160	40	2	7-6-1	1.5919
80	120	5	7-9-1	3.6296
120	80	5	7-8-1	2.3012
160	40	5	7-6-1	2.0505
80	120	8	7-8-1	3.9255
120	80	8	7-5-1	2.7544
160	40	8	7-7-1	2.6229
80	120	12	7-10-1	4.7103
120	80	12	7-8-1	3.1837
160	40	12	7-7-1	2.8713

Based on Tables 3.48 and 3.49, the networks learn the dynamics of the degradations better when they are trained by using more data points. The RMSE decreases by 33.18% in 5 flights ahead prediction using 160 data in comparison to 80 data points. It should be noted that the RMSE increases in higher erosion indices. The RMSE increases 25.19% when the network predicted 5 flights ahead turbine temperature in $EI = 1\%$ rather than $EI = 3\%$.

Table 3.49: Summary of the prediction errors for each scenario in presence of $EI = 3\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-11-1	3.8966
120	80	2	7-8-1	3.2514
160	40	2	7-8-1	2.4929
80	120	5	7-9-1	4.6264
120	80	5	7-6-1	3.7137
160	40	5	7-7-1	2.7411
80	120	8	7-10-1	6.4607
120	80	8	7-7-1	5.3786
160	40	8	7-7-1	4.3814

3.1.3 Concurrent Degradations

As mentioned in Section 2.6.5, it is also possible that both fouling in the compressor and erosion in the turbine occur at the same time in the gas turbine engine. These degradations were modelled in our Simulink model in Section 2.6.5 based on the work of Naeem [142] and equations (2.6.1)-(2.6.4). These data are used to train different NARX neural network structures. The trained networks are then fixed to be used in the testing phase to predict multi-flight ahead turbine temperature. The predictions are used for maintenance actions to decide whether the next flights will be safe or not.

3.1.3.1 FI=1% and EI = 1%

Turbine output temperatures are predicted in presence of 1% compressor fouling and 1% turbine erosion. Compressor fouling or turbine erosion do not occur in only one flight, but they occur over multiple flights. In this case the efficiency of the compressor reduces by 1% and the compressor mass flow rate reduces by 0.5% in over 200 flights while at the same time the efficiency of the turbine degrades by 1% and its mass flow rate increases by 0.5% due to the removal of materials from turbine part of the

gas turbine engine. Here 80 data points are used to train different NARX neural network structures to predict 2 flights ahead turbine temperature. The predictability of these networks were then validated by applying them to 120 data points. The error of prediction, standard deviation and RMSE for these networks are summarized in Table 3.50. The network with 9 hidden neurons has the lowest RMSE ($5.8402K$). Figure 3.57 shows the actual turbine temperatures versus predicted values. It can be seen that 60.83% of the turbine temperature data are within the prediction bounds.

Table 3.50: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.3017	3.9626	9.1918
6	7.7436	3.8228	8.6288
7	7.2098	3.7926	8.1391
8	6.1020	3.2729	6.9179
9	**3.3303**	**4.8177**	**5.8402**
10	5.8379	4.4998	7.3594
11	6.7557	3.7664	7.7271
12	6.4514	4.9517	8.1200
13	7.2042	4.2314	8.3461
14	4.3068	7.4609	8.5878
15	6.2519	6.2823	8.8445

In order to investigate the effect of training data on the performance of the network, 120 of the 200 available data points are now used in the training phase and the remaining 80 data points are used in the testing phase to predict the turbine temperature in 2 flights ahead. Table 3.51 shows that the lowest RMSE is $4.8784K$ which is 16.47% lower than in the case when the network was trained by using 80 data points. Figure 3.58 shows the predicted data and actual ones as well as the upper and the lower prediction bounds where 63.75% of the predicted data points are within these prediction bounds.

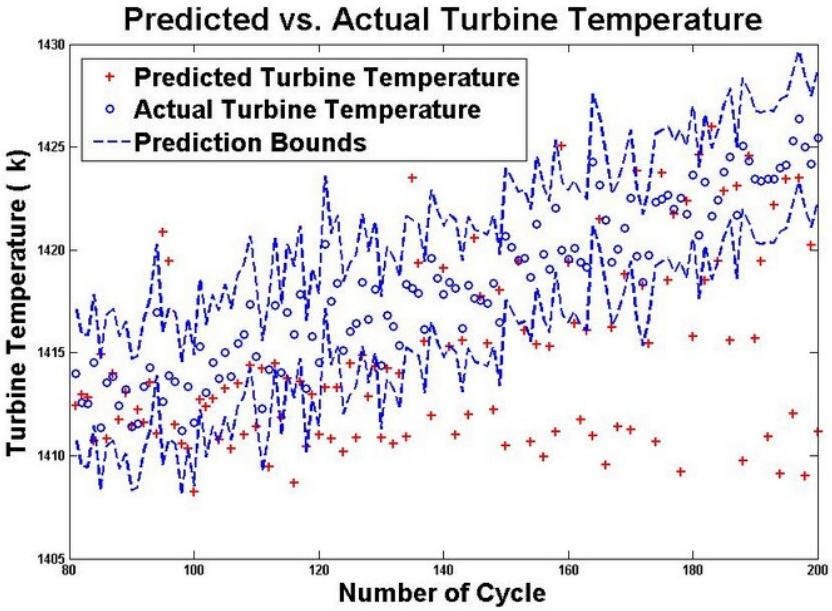

Figure 3.57: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

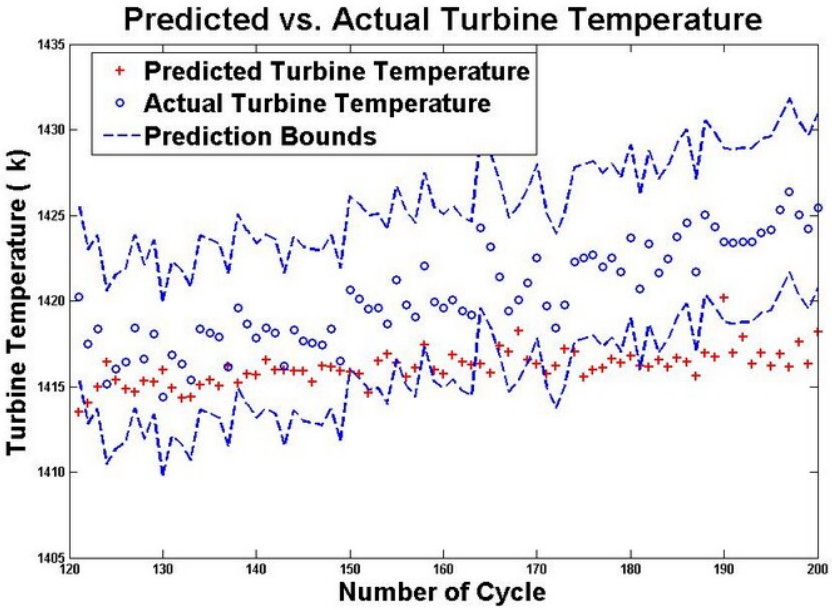

Figure 3.58: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 3.51: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	0.8823	7.6185	7.6220
6	6.6688	3.1138	7.3517
7	5.1691	4.0643	6.5599
8	5.2708	2.6336	5.8848
9	**4.1951**	**2.5057**	**4.8784**
10	4.5059	3.5062	5.6958
11	5.1631	2.7298	5.8323
12	5.3930	2.4173	5.9038
13	5.9038	3.2157	6.7937
14	6.1801	3.2562	6.9760
15	6.1183	3.9740	7.2821

Next, 80% of the entire data sets are used to train the networks. The trained networks are then used to predict 40 unseen data. Different NARX neural network structures are used and tested. The error results are presented in Table 3.52. When the network has 8 hidden neurons the RMSE is $2.3584K$. The predicted data and actual ones are depicted pointwise in Figure 3.59 for the network 7-8-1 where only 10% of the predicted data are outside the prediction intervals. The absolute difference between the actual and predicted data points are shown in Figure 3.60.

Next the 5 flights ahead turbine temperatures are predicted by using different NARX neural network structures. Different number of training and testing data are used in each case. Table 3.53 shows the results of the prediction when the neural networks are trained by using 80 data points. The RMSE of the prediction error decrease to $5.0015K$ when the network used 120 data points in the training phase as shown in Table 3.54. The lowest RMSE is achieved when 160 data points are used to train the network based on Table 3.55 for the network with 6 hidden neurons. Figure 3.61 shows the actual and predicted data points using 40% of the entire data

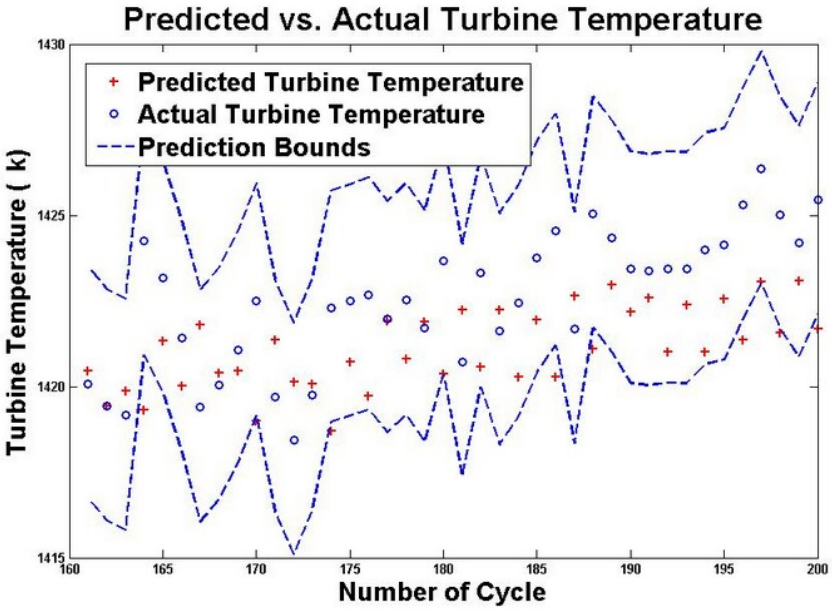

Figure 3.59: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

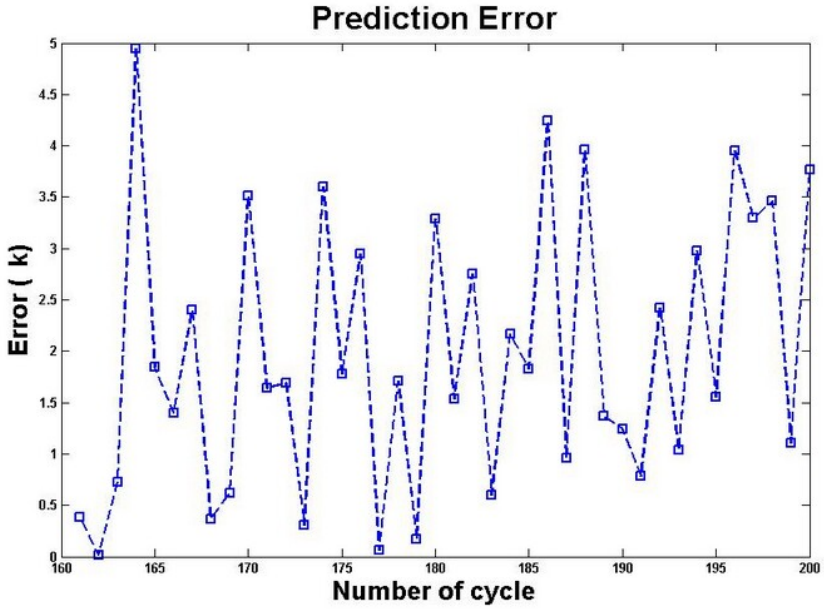

Figure 3.60: The prediction errors for 2 step ahead turbine temperature prediction using NARX 7-8-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 3.52: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	3.8129	2.0808	4.3313
6	2.9650	2.3024	3.7363
7	2.8115	2.0855	3.4850
8	**1.4211**	**1.9062**	**2.3584**
9	0.0816	2.7113	2.6785
10	2.9834	1.7731	3.4592
11	3.0683	2.0233	3.6614
12	2.5788	2.8366	3.8073
13	3.3585	2.0628	3.9279
14	3.5833	1.9401	4.0632
15	3.4893	2.5879	4.3250

sets in the training phase for the network 7-11-1 where 44.17% of the predicted data are within the prediction intervals. This value increases to 55% when 60% of the data are used in the training phase in the network 7-8-1 as depicted in Figure 3.62, and finally for the network 7-6-1 which is trained by using 80% of the available data points, 72.5% of the predicted data are between the upper and the lower prediction bounds as shown in Figure 3.63.

The NARX neural networks are trained with the available data sets and 8 flights ahead turbine output temperature is predicted for maintenance actions. Three different cases are assumed for this scenario. First, the number of hidden neurons in the NARX neural network structure is changed from 5 to 15. The networks are trained with 80 data points. The weights and biases remain fixed and the networks are used to predict 8 flights ahead turbine temperature. The results for prediction errors are tabulated in Table 3.56 where the network with 10 hidden neurons has the lowest RMSE. The actual and predicted data are shown in Figure 3.64. It can be seen that only 46.67% of the predicted data are within the prediction bounds.

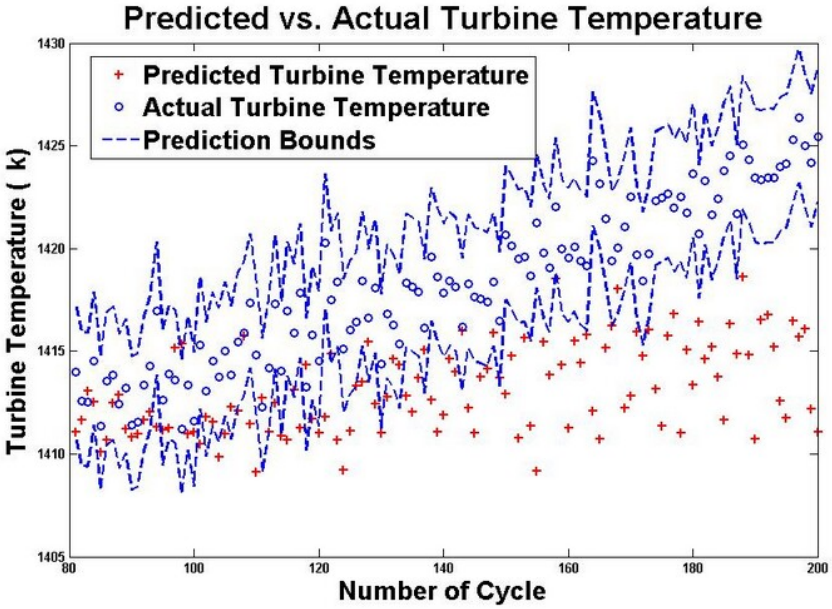

Figure 3.61: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

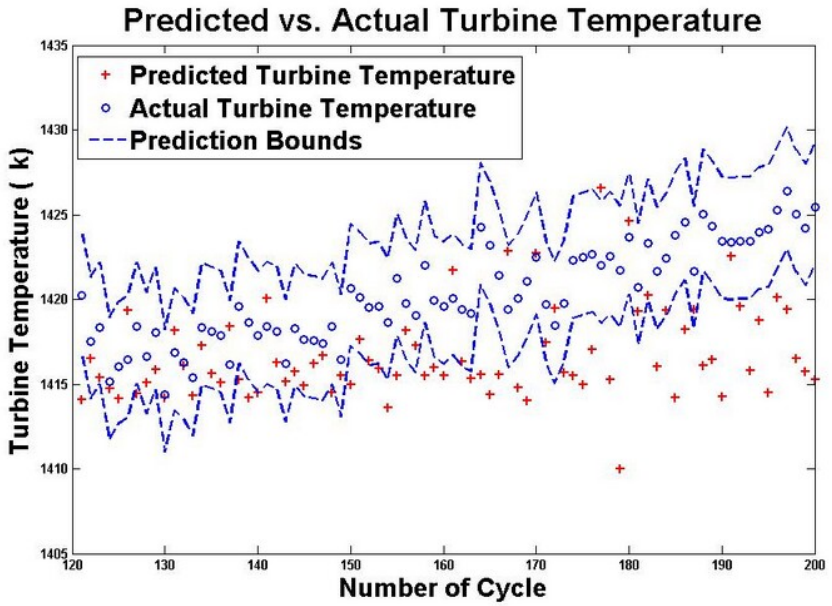

Figure 3.62: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

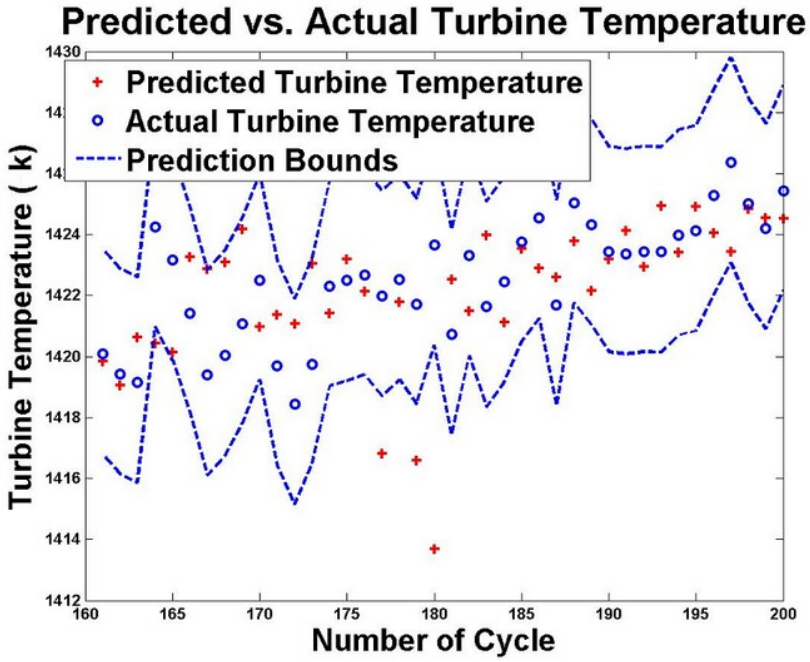

Figure 3.63: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-6-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

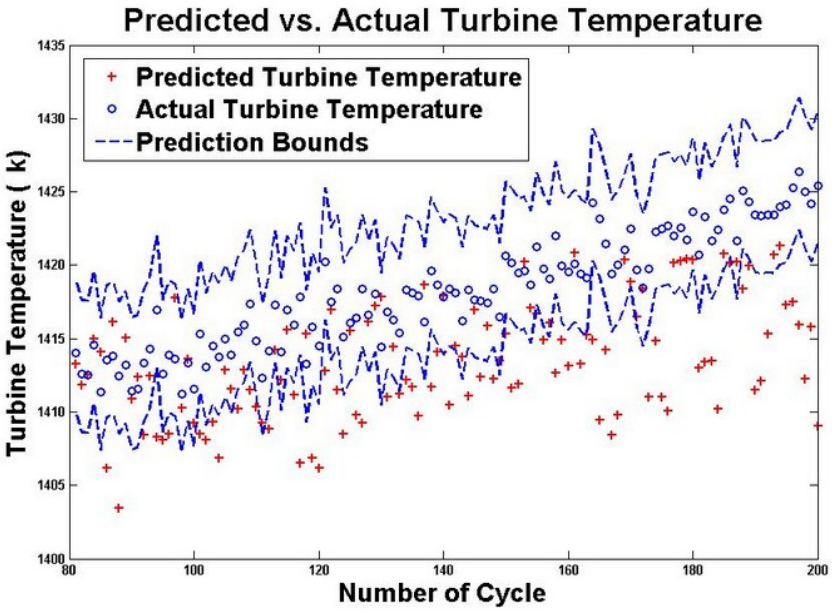

Figure 3.64: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 3.53: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	9.7739	4.8059	10.8827
6	8.3756	4.8678	9.6772
7	8.0872	5.1000	9.5497
8	7.7264	3.9586	8.6740
9	7.3201	3.7759	8.2294
10	6.8178	3.2949	7.5663
11	**5.0980**	**3.5872**	**6.2250**
12	5.9816	5.5637	8.1533
13	7.9605	3.8178	8.8217
14	8.2773	4.0087	9.1896
15	10.1040	5.7402	11.6089

When the training data increases to 120 data points, the RMSE decreases to $5.9867K$ for the network with 9 hidden neurons where 60% of the predicted data from this network are within the upper and the lower bounds as depicted in Figure 3.65. Table 3.57 summarizes the prediction error when the number of hidden neurons change from 5 to 15 using 120 data points during training phase and 80 data points during testing phase.

Data used in the training phase increases to 160 data points. Different NARX neural network structures are trained and their performance in 8 flight ahead turbine temperature prediction are evaluated by using 40 unseen data. The prediction error for these networks are shown in Table 3.58 where the network with 7 hidden neurons has the lowest RMSE. The actual and predicted data for the network 7-7-1 are shown pointwise in Figure 3.66 where 75% of the predicted data points are within the upper and the lower prediction intervals.

The applicability of the NARX neural network to predict 12 flights ahead turbine temperature in presence of 1% compressor fouling and 1% turbine erosion is also

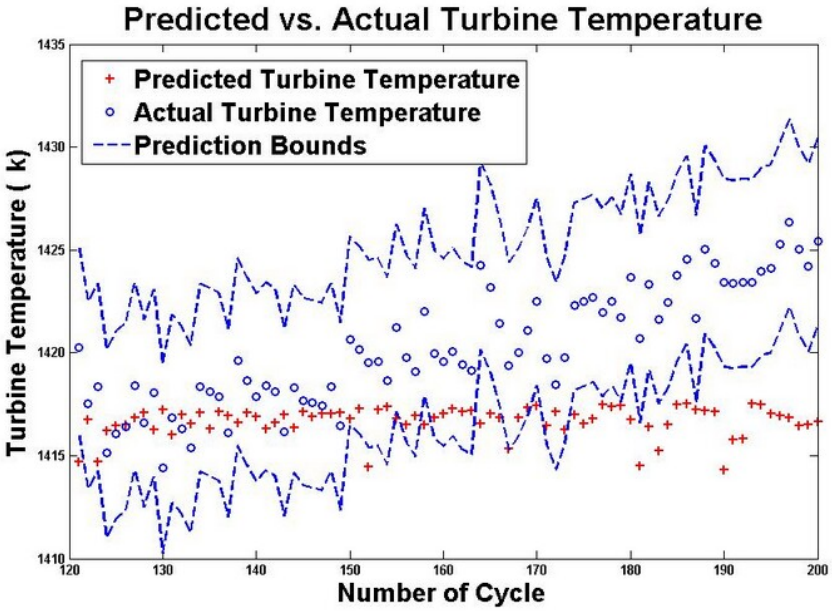

Figure 3.65: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

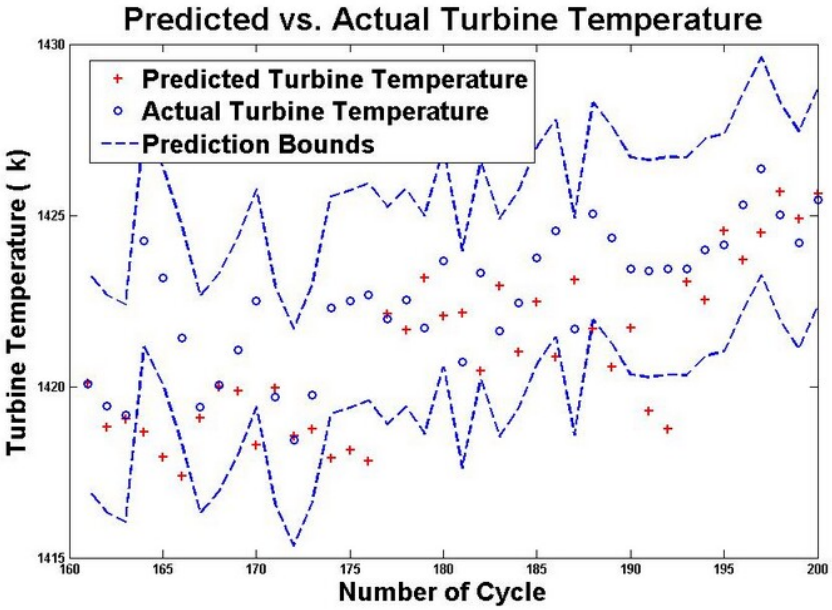

Figure 3.66: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-7-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 3.54: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	6.3820	3.0836	7.0795
6	5.7182	2.8829	6.3957
7	5.3805	2.4939	5.9238
8	**3.6317**	**3.4605**	**5.0015**
9	3.9005	3.2556	5.0676
10	4.9719	2.6836	5.6419
11	5.3756	2.8481	6.0752
12	5.9564	3.5605	6.9280
13	6.6438	3.3468	7.4297
14	6.3698	4.1517	7.5891
15	6.9829	3.6181	7.8541

investigated through three different cases. In the first case, the neural networks are trained by using 40% of 200 available data points and 120 data points are given as unseen inputs to the network to predict 12 flights ahead turbine temperature. The results of the prediction error, standard deviation and RMSE are tabulated in Table 3.59. Figure 3.67 shows the actual and predicted values for the 120 data points for the NARX network 7-11-1. It can be seen that only 43.33% of the predicted data points are within the prediction bounds.

In the second case, the trained data sets increase to 60% of the entire data sets which is equal to 120 data points. The remaining 80 data points are used in the evaluation section. Various NARX neural network structures are trained and tested. The results of the error are summarized in Table 3.60 where 61.25% of the prediction values for the network with 9 hidden neurons are within the upper and the lower prediction bounds as shown in Figure 3.68.

Finally, in the third case, 80% of the available data points are used in the training phase. The trained networks are then evaluated with 40 data points. Twelve flights

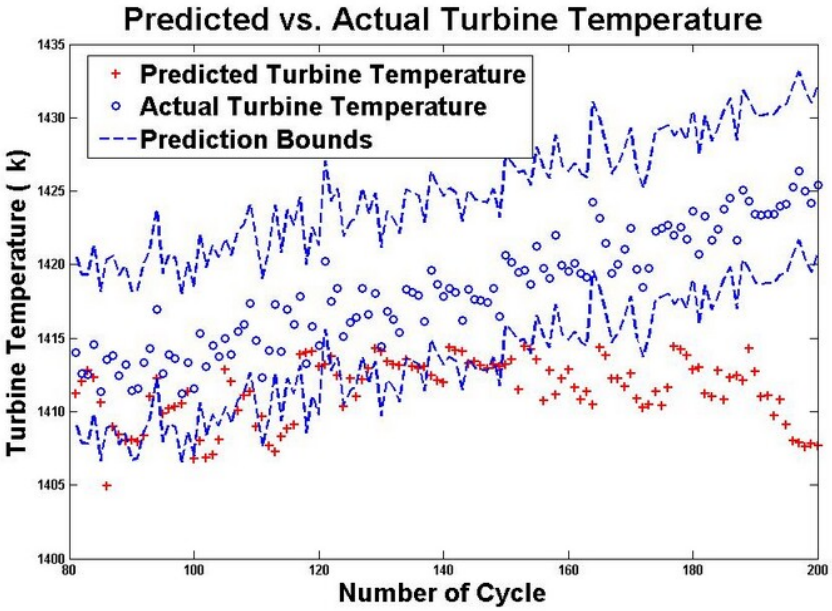

Figure 3.67: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

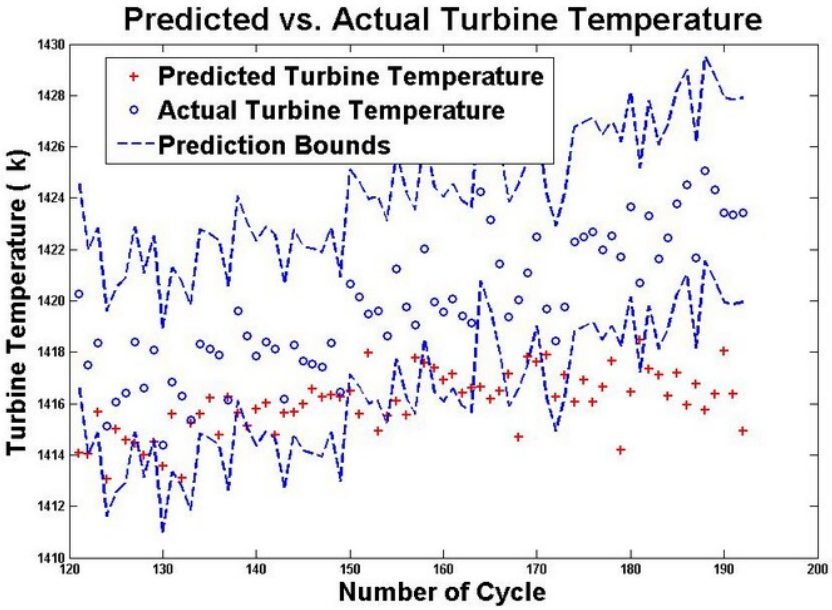

Figure 3.68: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 3.55: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.4139	2.4771	5.0463
6	**2.6101**	**2.0276**	**3.2896**
7	2.8509	1.9898	3.4623
8	3.2284	2.1775	3.8788
9	3.2171	2.6000	4.1159
10	3.5481	2.5050	4.3252
11	3.9460	2.5038	4.6566
12	4.1416	2.6496	4.8987
13	4.3552	2.7607	5.1380
14	4.7136	2.6410	5.3869
15	5.8907	2.4556	6.3702

ahead turbine temperatures are predicted and the errors which are the differences between the actual values and predicted ones are presented in Table 3.61. To consider the uncertainty in measurements, prediction bounds are depicted in Figure 3.69 for the network structure 7-9-1 where 87.5% of the predicted data points are within the prediction intervals.

Table 3.56: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	9.6906	3.9835	10.4711
6	7.7002	5.7434	9.5919
7	8.2901	6.2710	9.3175
8	6.2717	4.5585	7.7422
9	5.3383	4.4261	6.9228
10	**5.4807**	**3.4768**	**6.4827**
11	6.4709	6.5035	7.3515
12	7.4099	6.8401	8.3384
13	7.4855	4.2185	8.5837
14	6.1546	7.6630	9.8036
15	7.7428	8.2630	11.2986

Table 3.57: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.6967	7.0608	9.2132
6	8.5340	3.7060	8.9476
7	8.0572	7.9498	8.5739
8	7.3802	3.1176	8.0041
9	**4.5406**	**3.9263**	**5.9867**
10	5.4802	3.0096	6.2432
11	5.9711	3.0085	6.6778
12	6.3626	5.7908	6.9407
13	6.7704	2.7720	7.3093
14	7.0009	6.1994	8.1503
15	6.2315	6.8481	9.2272

Table 3.58: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.5891	3.4564	5.7191
6	4.4224	2.0615	4.8684
7	**2.3273**	**2.7251**	**3.5577**
8	3.5367	2.1896	4.1452
9	3.4855	3.4350	4.2344
10	4.2906	2.3949	4.8991
11	4.4789	3.5385	5.1326
12	2.4464	4.6657	5.2162
13	5.0567	2.1031	5.4665
14	5.0104	3.8606	5.7517
15	5.1761	3.4382	5.7086

Table 3.59: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	10.8399	4.2902	11.6514
6	10.2835	4.8854	11.3762
7	9.3991	5.2037	10.7329
8	8.5830	4.4981	9.6815
9	8.6199	5.9633	9.4805
10	7.6801	3.9676	8.6369
11	**6.8235**	**5.0442**	**7.9234**
12	8.5621	5.8179	9.3683
13	8.5976	6.1109	9.5224
14	8.1590	6.3225	9.2248
15	6.6419	6.8566	9.5256

Table 3.60: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.9698	7.7913	9.3883
6	8.6562	6.6796	9.0560
7	7.6308	6.3876	8.3394
8	7.0147	2.5152	7.4461
9	**6.0475**	**4.5066**	**6.9783**
10	6.7871	6.5145	7.2318
11	6.3269	4.8851	7.9727
12	5.6465	5.7733	8.0468
13	7.8228	5.8339	8.3136
14	5.6144	7.5066	9.3321
15	8.8204	5.1000	9.7147

Table 3.61: A 12 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.7213	5.4567	6.2130
6	0.3881	5.8408	5.7722
7	4.8530	4.7592	5.5636
8	3.6288	3.0608	4.7198
9	**3.5002**	**3.2657**	**4.1524**
10	4.1451	3.5034	4.8244
11	4.4796	3.4286	5.0794
12	3.1645	4.0515	5.0964
13	4.8074	4.5508	5.4256
14	3.0103	5.0557	5.8233
15	5.8217	5.7911	6.4394

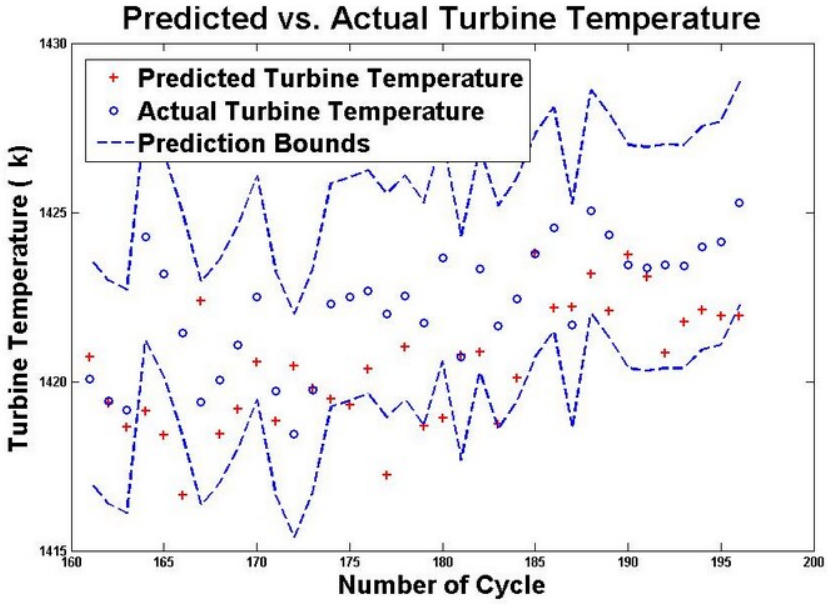

Figure 3.69: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

3.1.3.2 FI=3% and EI = 2%

It is assumed for this scenario that both compressor fouling and turbine erosion occur at the same time in the gas turbine engine with different percentages. The compressor degrades by 3% and the turbine degrades by 2% in 200 flights. The compressor efficiency decreases 3% and its mass flow rate decreases by the amount of 1.5% due to fouling. Turbine efficiency decreases 2% and its mass flow rate increases 1%. The data are obtained by using the engine model which is simulated as mentioned in Section 2.5 and equations (2.6.1)-(2.6.4). These data are used to train various NARX neural networks. The trained networks are then used to predict multi-flights ahead turbine temperature and the prediction errors are compared together through various simulations.

In the first case, different NARX neural network structures are trained by using 80 data points to predict 2 flights ahead turbine temperature. These networks are then used to predict the remaining 120 data points. The actual values and predicted ones are compared together and the errors are summarized in Table 3.62. The network structure 7-10-1 has the lowest RMSE ($7.9045K$). To overcome the uncertainty in measurements, the prediction intervals are depicted as shown in Figure 3.70 where 49.16% of the predicted data are within the prediction bounds using the network 7-10-1.

Next, different NARX neural network structures are trained by using 120 data points. The number of hidden neurons increases from 5 to 15. The performance of these networks are evaluated to predict 2 flights ahead turbine temperature. The results are presented in Table 3.63. The actual turbine temperatures and predicted values for the network 7-9-1 are shown pointwise in Figure 3.71. From Figure 3.71, 56.25% of the predicted data are within the upper and the lower prediction bounds.

By increasing the number of the training data points to 80% of the entire available

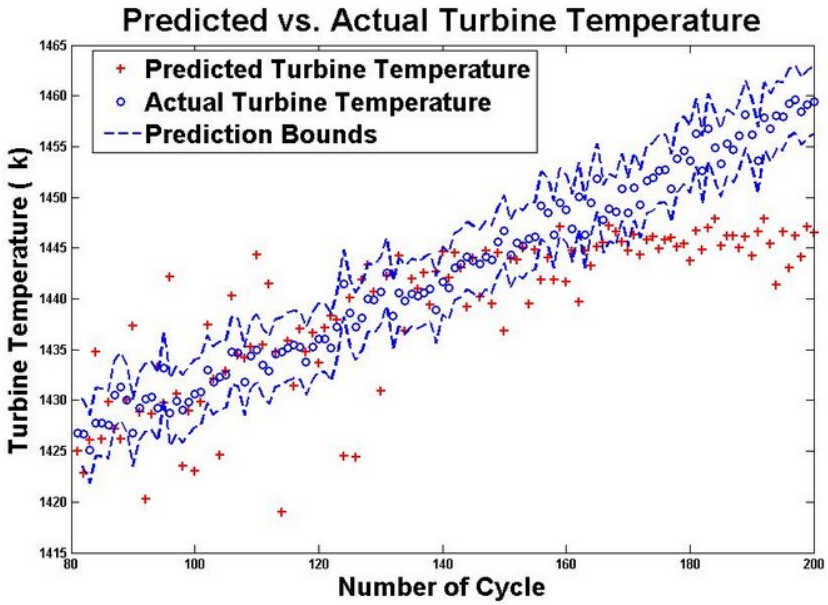

Figure 3.70: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$.

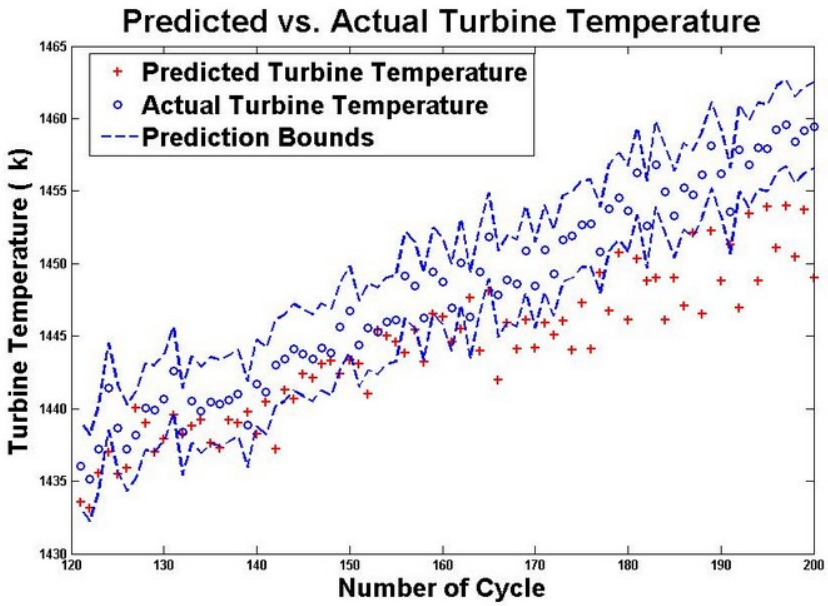

Figure 3.71: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 3.62: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.7832	8.3773	12.1136
6	9.4761	8.6660	12.8168
7	2.4850	10.4216	10.6714
8	4.2685	9.0239	9.9484
9	1.8230	8.8295	8.9796
10	**6.7517**	**6.1276**	**7.9045**
11	4.4103	7.9123	9.0296
12	4.0022	8.2370	9.1269
13	9.4333	5.1369	10.7311
14	10.5269	5.7626	11.9894
15	12.3957	4.9337	13.3339

data, the lowest RMSE is achieved when the network has the structure of 7-9-1 as shown in Table 3.64. The mean, standard deviation and RMSE of the prediction are $1.8772K$, $1.7692K$, and $2.5643K$, respectively. Actual and predicted values are shown pointwise in Figure 3.72 where 85% of the predicted data are within the prediction intervals.

Next the 5 flights ahead turbine temperatures are now predicted by using the NARX neural networks under different cases. First, 80 data points are used to train the network, and 120 data points are given to the network as the input and 5 flights ahead turbine temperatures are predicted. The results of the error in the prediction are tabulated in Table 3.65. Next, training data increased to 120 data points and 80 data are predicted as shown in Table 3.66, and finally 160 data points are used in the training phase and 40 data points in the testing phase. Comparing Tables 3.65 and 3.67 the RMSE decreases 67.88% when the training data increases 50%.

The actual and predicted values along with prediction intervals are depicted in Figure 3.73 for the network structure 7-9-1 when this network is trained by using

Table 3.63: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.6341	4.8626	9.8943
6	8.1353	4.4920	9.2795
7	7.1066	3.3641	7.8537
8	3.7544	6.3696	7.3594
9	**3.3791**	**4.6264**	**5.7057**
10	0.8893	7.3926	7.3999
11	7.0577	4.6374	8.4290
12	8.4140	4.7490	9.6471
13	9.7202	6.3285	11.0689
14	10.4583	8.3641	11.7383
15	11.7438	7.4376	12.9273

40% of the entire data sets. Only 45% of the predicted data are within the prediction bounds. This value increases to 62.5% in Figure 3.74 which shows the predicted values for the network structure 7-10-1 which was trained by using 60% of the available data points, and finally as shown in Figure 3.75, 85% of the predicted data are inside the prediction intervals when the network 7-9-1 is trained by using 80% of the total data points.

The applicability of the NARX neural networks in 8 flights ahead turbine temperatures prediction is also investigated. Total data available to use in the training and the testing phases are 200 points. First, 80 data points are used to train neural networks. The number of hidden neurons is increased from 5 to 15, and different NARX neural networks are trained. These networks are then evaluated using 120 data points. The predicted outputs are compared to the actual ones, and the errors are presented in Table 3.68. The network with 10 hidden neurons has the lowest RMSE equal to $14.4070K$. The actual and predicted data for this network in the testing phase are shown pointwise in Figure 3.76 where only 35.83% of the predicted

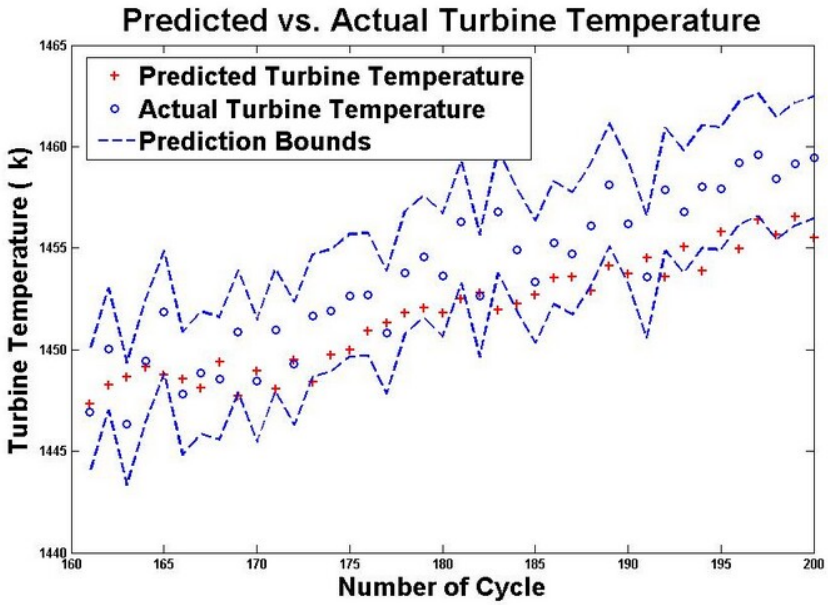

Figure 3.72: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$.

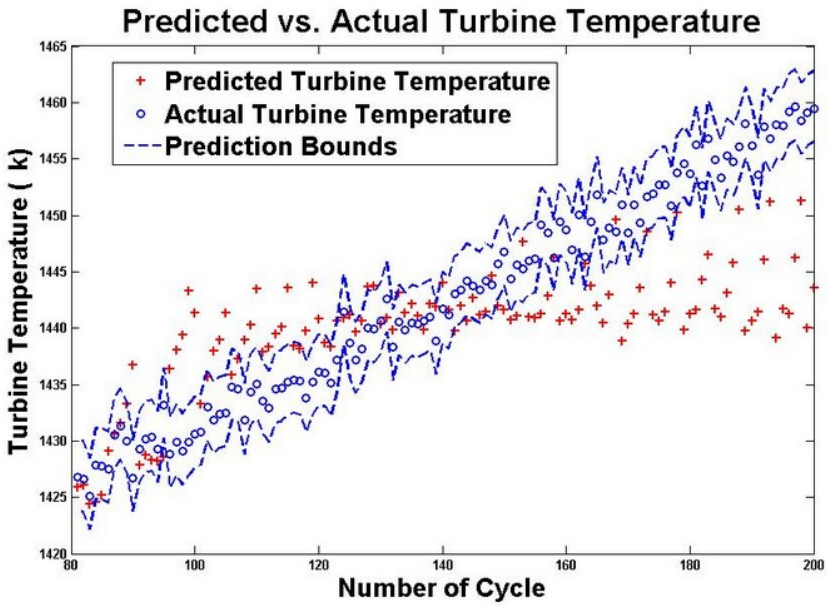

Figure 3.73: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$.

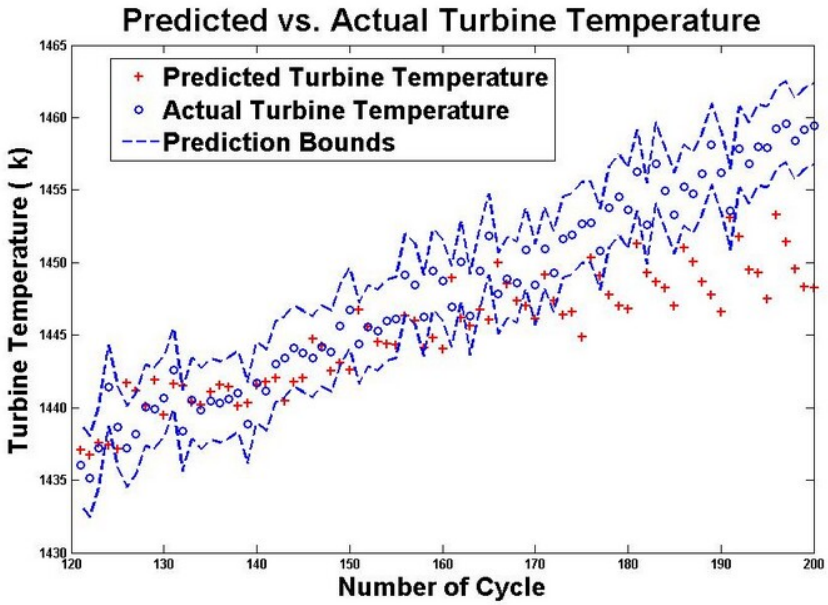

Figure 3.74: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$.

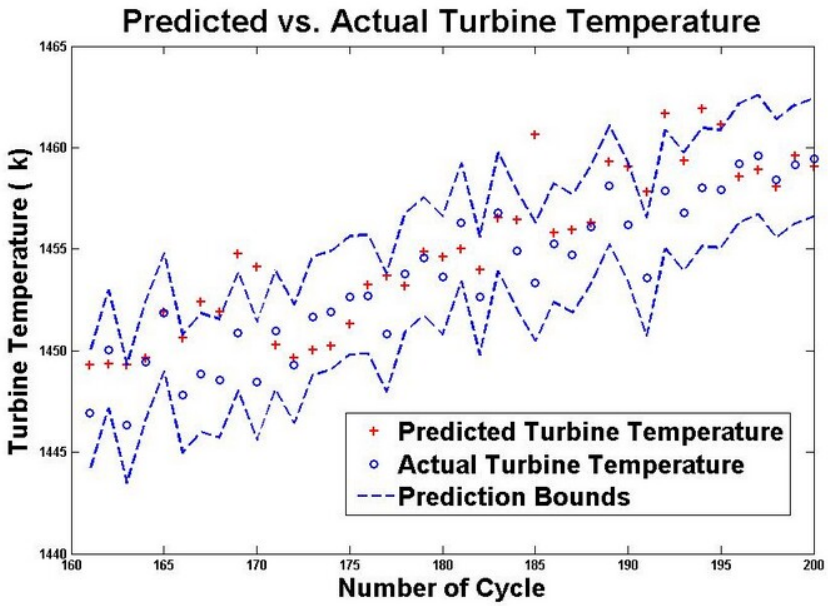

Figure 3.75: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 3.64: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.4775	2.4095	5.0704
6	3.0607	2.6317	4.0150
7	2.6263	2.6601	3.7144
8	2.1130	2.7633	3.4511
9	**1.8772**	**1.7692**	**2.5643**
10	1.9209	2.3088	2.9811
11	2.5773	2.0748	3.2924
12	1.3583	3.2259	3.4628
13	2.3422	2.7009	3.5494
14	3.4391	2.4994	4.2330
15	3.8659	2.6478	4.6670

values are within the prediction bounds.

Training data increases to 120 data points as done previously where different NARX neural networks are trained and tested by the remaining 80 data points to find the optimal structure. Summary of the prediction error for these neural networks can be seen in Table 3.69. The actual and predicted data turbine temperature for the NARX neural network 7-11-1 are depicted in Figure 3.77 where 56.25% of the predicted data are within the upper and the lower prediction intervals.

When the training data increases to 160 data points, the RMSE decreases to $4.1737K$ as presented in Table 3.70 for the network with the structure of 7-8-1. The actual and predicted values for this network are shown in Figure 3.78 where 85% of the predicted data points are within the upper and the lower prediction intervals.

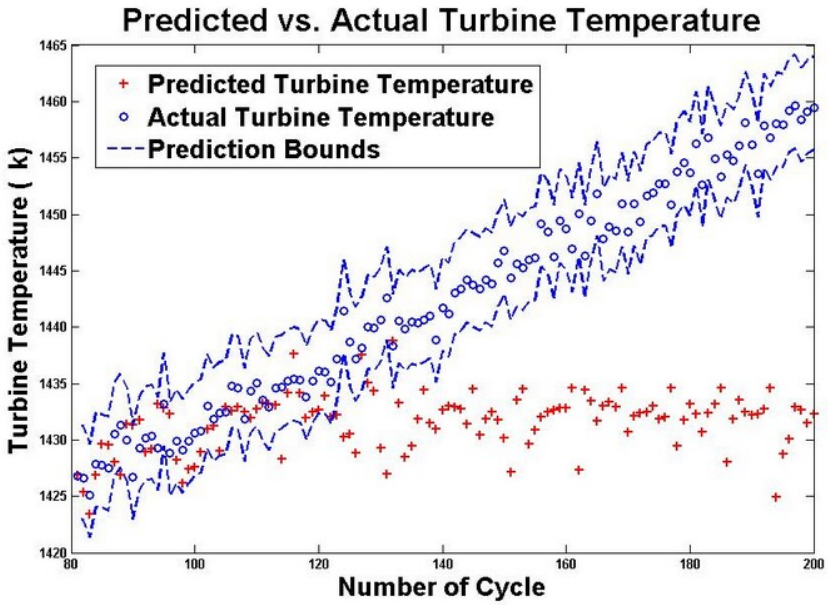

Figure 3.76: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$.

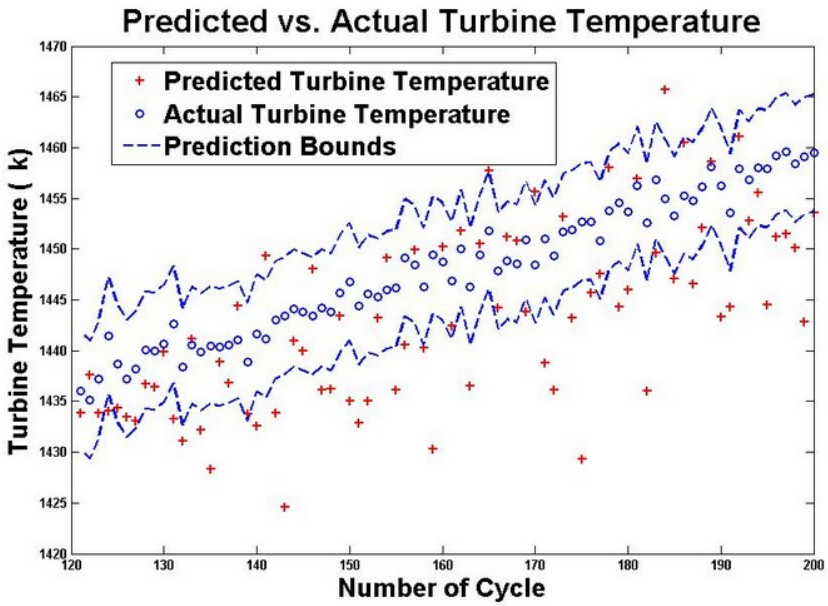

Figure 3.77: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 3.65: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	12.9333	8.2926	15.3449
6	11.5804	7.3566	13.7030
7	8.7180	9.4226	12.8081
8	10.4135	7.6127	12.8806
9	**7.0833**	**7.5034**	**10.2959**
10	8.1156	10.1260	12.9439
11	10.8225	9.7591	14.5455
12	13.8122	7.9806	15.9353
13	14.4565	9.7066	17.3903
14	15.6480	11.5768	19.4362
15	16.7434	9.2267	19.0988

Table 3.66: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	9.8367	5.9501	11.4770
6	9.4641	5.3497	10.8550
7	9.2584	5.0127	10.5134
8	6.1198	8.3927	10.3445
9	5.4717	6.1817	8.2265
10	**6.3297**	**4.1259**	**7.5415**
11	7.3068	4.6741	8.6581
12	8.4912	5.8153	10.2711
13	10.5707	5.7139	11.9992
14	11.8000	6.3364	13.3749
15	10.9266	8.3183	13.7011

Table 3.67: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.0257	2.9173	5.7927
6	5.0799	2.8934	5.8282
7	3.3701	2.6276	4.2531
8	2.6127	2.9699	3.9276
9	**2.4046**	**2.2983**	**3.3064**
10	3.0942	2.6386	4.0450
11	3.4171	2.5782	4.2612
12	3.2607	3.1782	4.5256
13	3.6007	3.0292	4.6810
14	2.9373	3.8562	4.8090
15	4.1974	3.3073	5.3182

Table 3.68: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	20.5033	10.4985	23.0149
6	19.1627	10.3366	21.7524
7	17.3023	10.7341	20.3379
8	15.5348	9.9133	18.4061
9	13.7050	9.1191	16.4406
10	**10.9484**	**9.4038**	**14.4070**
11	11.6201	9.2355	14.8193
12	13.1722	8.8732	15.8614
13	14.1578	9.1939	16.8602
14	15.2158	9.5227	17.9289
15	0.3043	20.5372	20.4537

Table 3.69: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	13.7393	8.4316	16.0926
6	14.6826	6.5364	16.0552
7	13.4530	6.4243	14.8909
8	13.0830	6.8458	14.7460
9	11.7161	6.3849	13.3238
10	10.9097	6.1620	12.5107
11	**8.8926**	**7.7130**	**10.5503**
12	9.3311	5.8667	11.0026
13	6.7579	8.8590	11.0982
14	9.8434	5.4255	11.2232
15	12.2375	6.7388	13.9500

Table 3.70: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.5386	5.3098	6.4310
6	5.0507	4.6526	6.2062
7	4.4423	3.6802	5.7392
8	**1.9276**	**3.7491**	**4.1737**
9	3.1214	3.4528	4.6225
10	3.6223	3.6018	4.6186
11	4.5854	4.0130	5.4659
12	4.6213	3.4376	5.7340
13	3.9013	4.4040	5.8421
14	4.5122	4.2494	6.1616
15	4.7162	4.3268	6.3636

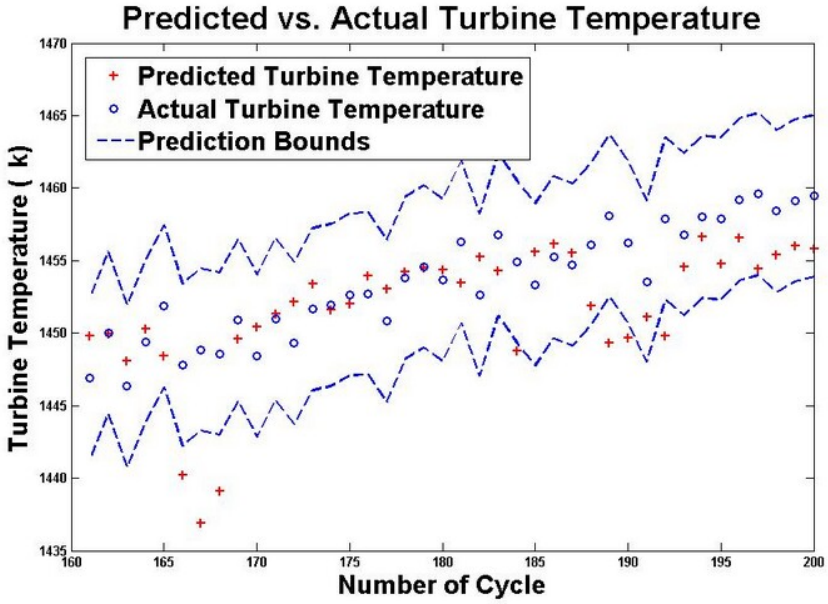

Figure 3.78: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$.

3.1.3.3 FI=2% and EI = 3%

In this section, it is assumed that the compressor degrades by the amount of 2% due to fouling and at the same time the turbine degrades by the amount of 3% due to erosion in 200 flights. The compressor efficiency decreases 2% and its mass flow rate decreases 1% and the turbine efficiency degrades 3% and its mass flow rate increases 1.5% because of these deteriorations.

The data generated in our Simulink model described in Section 2.6.5 are used to train and evaluate NARX neural networks. Different NARX neural networks are trained and tested by using various percentages of the available data. These networks are then used to predict multi-flights ahead turbine temperatures. First, different NARX neural network structures are trained by using 80 data points. The performance of these networks on 2 flights ahead turbine temperature prediction are then validated by 120 data points. The results of the prediction error for these networks are tabulated in Table 3.71. As shown in Table 3.71, the network with 8 hidden neurons has the lowest RMSE. The actual and predicted turbine temperatures along with the prediction intervals for this network are depicted in Figure 3.79 where 70.83% of the predicted data are within the prediction bounds.

When the training data increases to 120 data points the RMSE decreases to $5.4616K$ as shown in Table 3.72. The actual and predicted data points are shown in Figure 3.80 for the network structure 7-10-1 where 65% of the predicted data are within the prediction intervals.

The training data increases to 160 data points and different NARX neural network structures are trained. The weights and biases for these networks stay fixed and these networks are then evaluated by using 40 data points. The results of the prediction error for these networks are summarized in Table 3.73. The actual and predicted turbine temperatures are shown in Figure 3.81 for the network with 10 hidden neurons

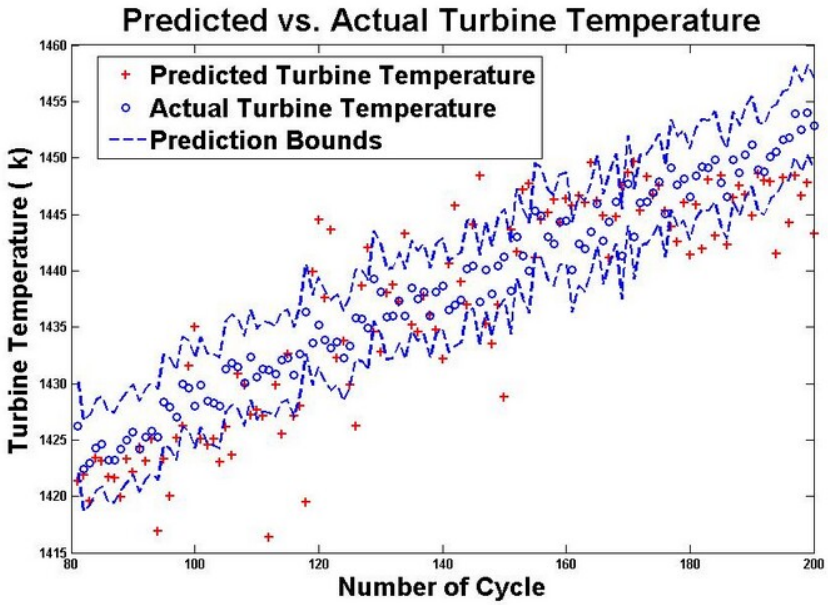

Figure 3.79: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-8-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

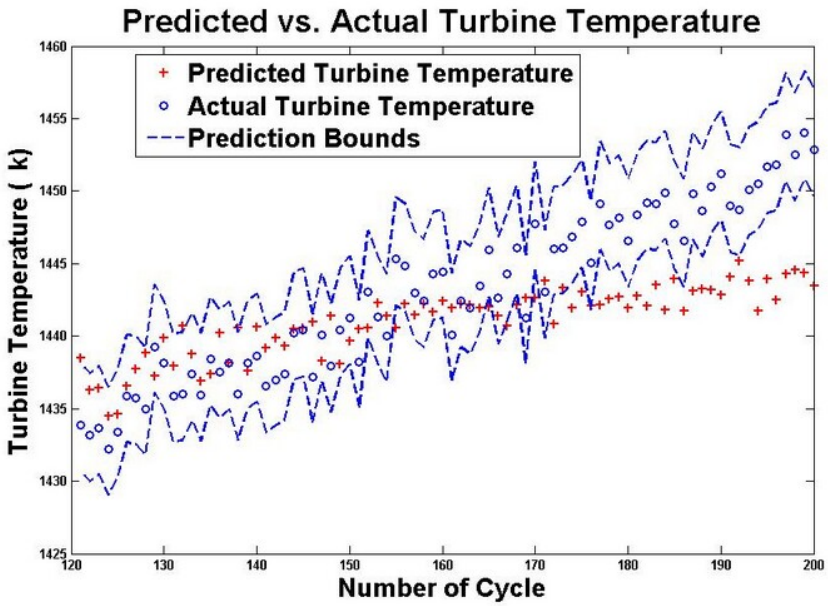

Figure 3.80: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 3.71: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	8.4260	6.4539	10.5973
6	6.8734	4.5040	8.2073
7	5.6363	4.8893	7.4481
8	**3.8914**	**5.4676**	**6.6924**
9	6.3480	3.8079	7.3902
10	6.9473	4.8999	8.4837
11	8.0732	4.1330	9.0579
12	7.0934	5.8312	9.1594
13	9.3818	6.1268	10.2390
14	8.7499	5.7920	10.4732
15	8.2973	6.5380	10.5384

where 87.5% of the predicted data are within the upper and the lower prediction bounds. The errors for the 40 data points used in the testing phase are shown in Figure 3.82.

Next the 5 flights ahead turbine temperatures are also predicted by using different NARX neural network structures with different numbers of training and testing data points. Table 3.74 shows the prediction error for the networks which are trained by using 80 data points and evaluated by using 120 data points. Table 3.75 summarizes the results when the networks are trained by using 120 data points. The performance of these networks are then evaluated by the remaining 80 data points, and finally the networks are trained with 160 data points and tested with 40 ones are shown in Table 3.76.

The actual and predicted turbine temperatures are depicted in Figures 3.83-3.85 for the three cases mentioned previously where 45% of the predicted data are within the prediction bounds when the network structure 7-11-1 is trained by using 80 data points. This value increases to 56.25% as shown in Figure 3.84 when the network

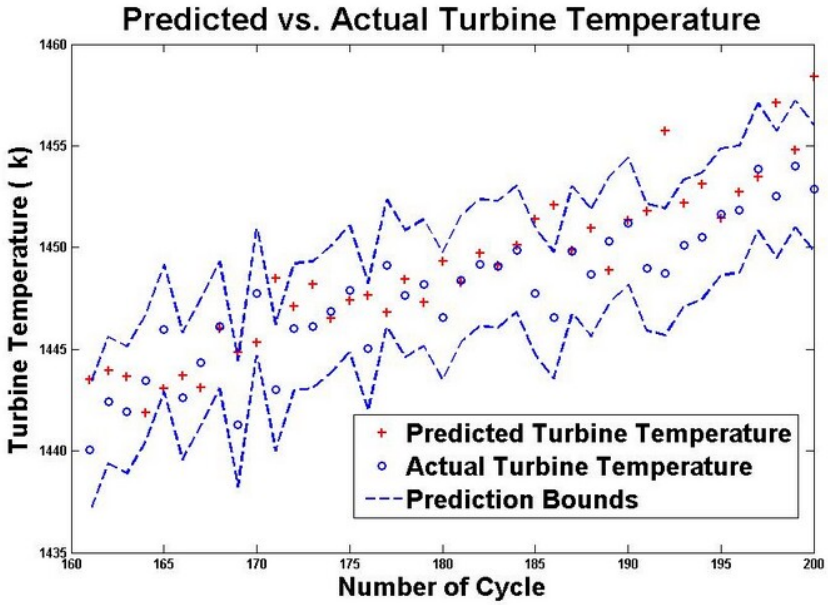

Figure 3.81: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

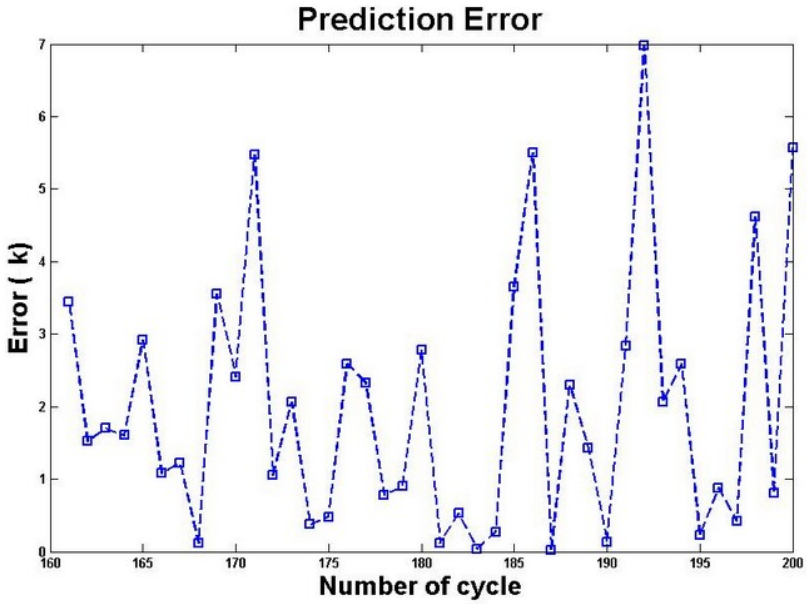

Figure 3.82: The prediction errors for the 2 step ahead turbine temperature when $FI = 2\%$ and $EI = 3\%$ using NARX 7-10-1 trained with 80% of the available data.

Table 3.72: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.7504	5.5926	7.3111
6	5.5321	3.4489	6.5077
7	4.8718	3.9168	6.2357
8	5.1952	2.9841	5.9820
9	3.1844	3.4492	4.6785
10	**2.0377**	**3.9941**	**5.4616**
11	4.3515	3.2026	5.3911
12	3.3814	4.4361	5.5557
13	4.7695	5.7532	7.4454
14	6.7818	4.6186	8.1888
15	7.6087	5.1869	9.1902

7-10-1 is trained with 120 data points, and finally 75% of the predicted data are within the prediction bounds as depicted in Figure 3.85 where the network is trained by using 160 available data points.

Next, 8 flights ahead turbine temperatures are predicted by using the NARX neural network where 80 data are used to train different network structures and these networks are then tested by using 120 remaining data. Note that d_u and d_y are both set to 3 and the number of hidden neurons is changed from 5 to 15. The results of the prediction error are tabulated in Table 3.77. The actual and predicted turbine temperatures along with their prediction bounds for the network with 12 hidden neurons are depicted in Figure 3.86 where only 34.1% of the predicted data are within the prediction bounds.

The results of the prediction error when the networks are trained by using 120 data points and tested with 80 data are summarized in Table 3.78. Figure 3.87 shows that 43.75% of the predicted data for the network structure 7-10-1 are within the prediction intervals.

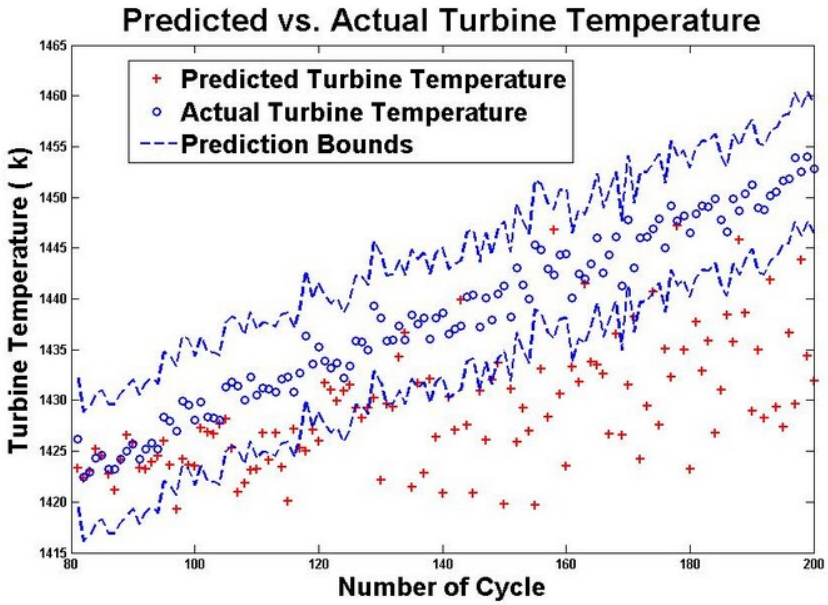

Figure 3.83: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

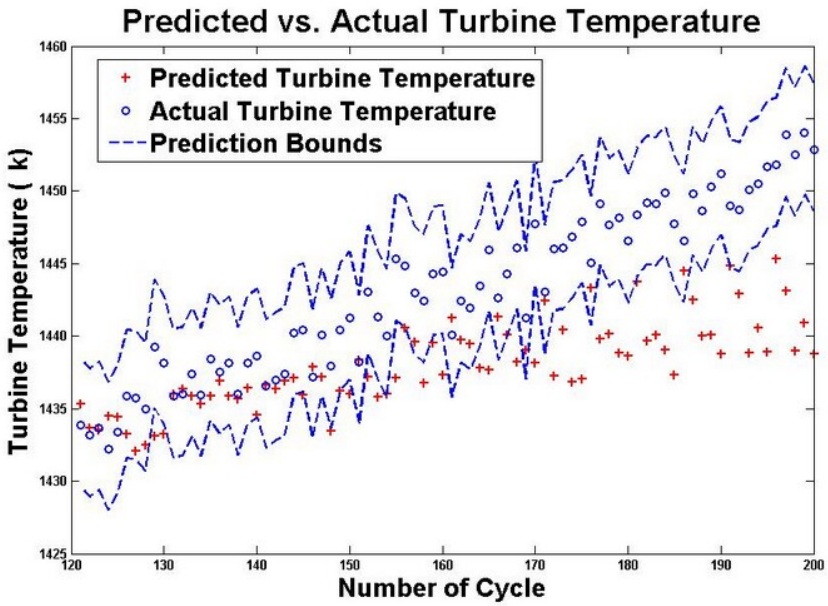

Figure 3.84: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

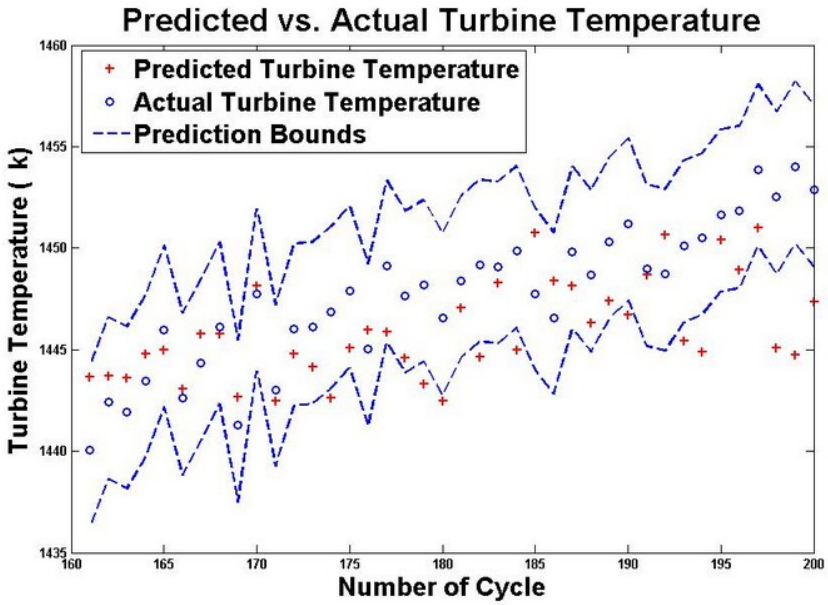

Figure 3.85: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

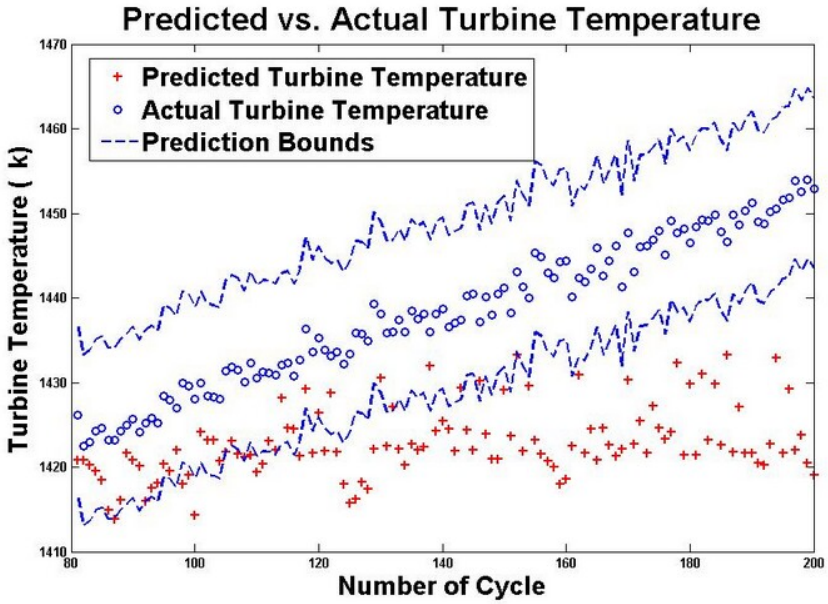

Figure 3.86: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-12-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 3.73: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.1778	2.8485	5.0364
6	3.8066	2.6179	4.6014
7	3.1913	2.5530	4.0669
8	2.5159	2.4166	3.4676
9	1.6127	2.1423	2.6600
10	**1.2566**	**2.3534**	**2.6419**
11	1.6684	2.1088	2.6682
12	2.5556	2.3510	3.4525
13	3.1783	2.9659	4.3218
14	3.9664	2.9941	4.9470
15	5.7980	2.9014	6.4672

Different NARX neural network structures are trained by using 160 data points. The weights and biases remain fixed and the performance of the networks are evaluated by using 40 unseen data. The prediction errors are shown in Table 3.79 where the network with 10 hidden neurons has the lowest RMSE. The actual and predicted turbine temperatures for this network are shown in Figure 3.88 where 82.5% of the predicted data are within the upper and the lower prediction bounds.

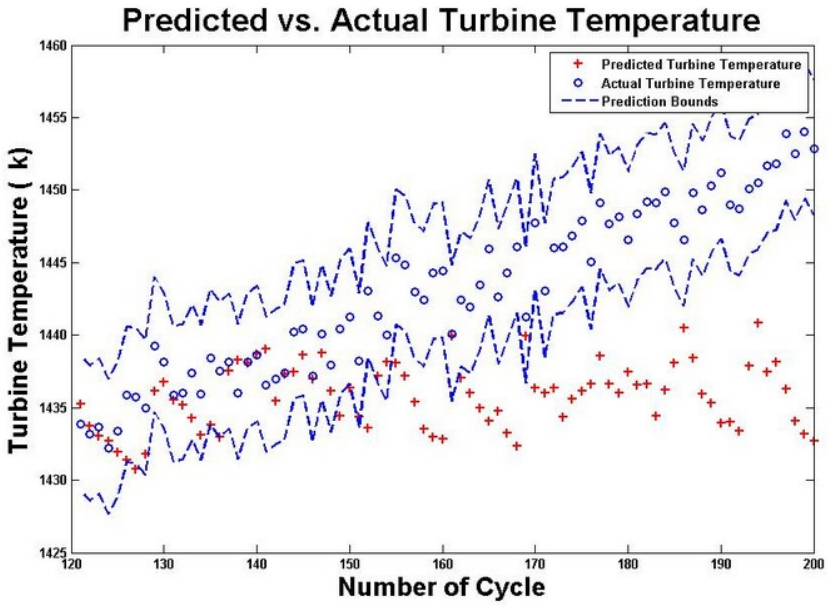

Figure 3.87: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

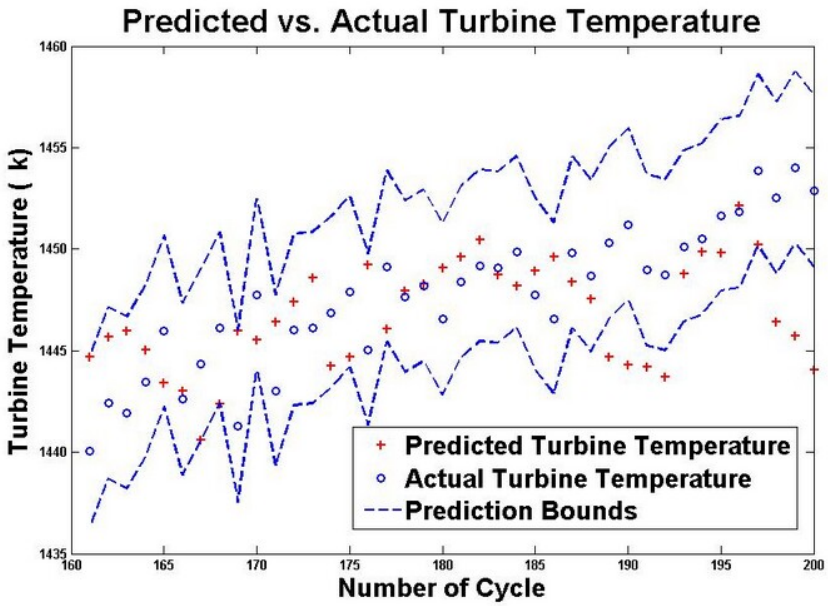

Figure 3.88: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 3.74: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	20.5303	11.1099	23.3215
6	17.0988	8.4411	19.0533
7	14.2158	10.0237	17.3703
8	14.3510	8.3092	16.5656
9	7.4152	11.0564	13.2744
10	9.8186	7.7857	12.5107
11	**9.1387**	**7.1012**	**11.5552**
12	10.6488	8.3867	13.5332
13	12.4504	7.4511	14.4937
14	12.6187	10.1850	16.1896
15	13.8096	9.3741	16.6687

Table 3.75: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	9.8423	5.9161	11.4645
6	9.6014	5.9797	11.2914
7	8.0704	5.7498	9.8883
8	7.8921	5.0527	9.3539
9	6.0753	4.6456	7.6303
10	**5.0387**	**4.1620**	**6.5188**
11	6.6337	4.3334	7.9088
12	6.8417	5.3748	8.6797
13	7.7636	6.3421	9.4050
14	9.1302	6.8936	11.4144
15	11.7492	8.7164	13.0503

Table 3.76: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	4.1628	3.0305	5.1267
6	3.4165	3.0086	4.5275
7	2.8283	3.0231	4.1122
8	2.2921	3.1147	3.8357
9	2.6027	2.7176	3.7383
10	**1.7818**	**2.9282**	**3.3963**
11	2.3929	2.6626	3.5550
12	2.1137	3.0881	3.7102
13	2.6638	2.9119	3.9196
14	2.6922	3.3402	4.2575
15	2.8464	3.1184	4.1932

Table 3.77: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	19.0451	11.9876	22.4771
6	18.7903	10.2153	21.3672
7	18.4043	8.9284	20.4394
8	18.2976	9.6353	20.6608
9	17.4263	9.0914	19.6378
10	17.4289	8.5985	19.4187
11	15.9244	9.3938	18.4687
12	**15.3787**	**7.8746**	**17.2626**
13	15.4387	9.1972	17.9510
14	16.4756	9.2509	18.8762
15	17.5003	8.6372	19.4998

Table 3.78: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	11.8179	5.6414	13.0801
6	11.8256	4.8682	12.7768
7	11.1891	5.7642	12.5701
8	9.0959	5.3308	10.5261
9	7.7616	6.5899	10.1551
10	**7.3068**	**5.8157**	**9.3160**
11	8.7576	5.2319	10.1846
12	9.7501	6.0891	11.4751
13	10.8335	6.3823	12.5535
14	11.1948	5.9115	12.6425
15	11.6558	7.8005	13.9980

Table 3.79: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	6.6548	3.8236	7.6512
6	5.1103	3.3844	6.1060
7	3.8866	3.5173	5.2123
8	3.8780	3.3448	5.0938
9	3.1339	3.2597	4.4923
10	**3.3341**	**3.0063**	**4.4641**
11	3.5776	3.6308	5.0648
12	3.9504	3.3291	5.1392
13	4.4922	3.8307	5.8726
14	5.0070	3.6628	6.1766
15	5.2447	4.0049	6.5685

3.1.3.4 FI=3% and EI = 3%

This is the most severe case of the others mentioned previously where both compressor and turbine efficiency in the gas turbine engine degrades by the amount of 3%. Fouling in the compressor decreases the amount of mass flow rate by 1.5% and at the same time the mass flow rate in the turbine increases by 1.5% due to the removal of the material in this section of the gas turbine engine. The applicability of the NARX neural network to predict the turbine output temperature is investigated by using different NARX structures which are trained by various numbers of available data points where 40% of the 200 available data sets are used in the training phase. Different NARX network structures are trained by changing the number of hidden neurons and their performance in the testing phase are compared together in Table 3.80. The NARX neural network structure 7-11-1 has the lowest RMSE. Actual and predicted values for this network in the testing phase are shown pointwise in Figure 3.89 where 68.33% of the predicted turbine temperatures are within the prediction bounds.

Table 3.80: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	11.8740	8.3382	14.4892
6	11.9577	9.0866	14.9955
7	11.1680	10.8334	14.2162
8	8.0708	8.9676	12.0368
9	7.9584	7.5348	10.9379
10	7.5474	6.6664	10.0516
11	**2.9696**	**8.0453**	**8.5444**
12	6.7463	8.1242	10.5340
13	8.8257	6.6908	11.0583
14	3.2469	10.9916	11.4171
15	7.0267	12.7429	14.5052

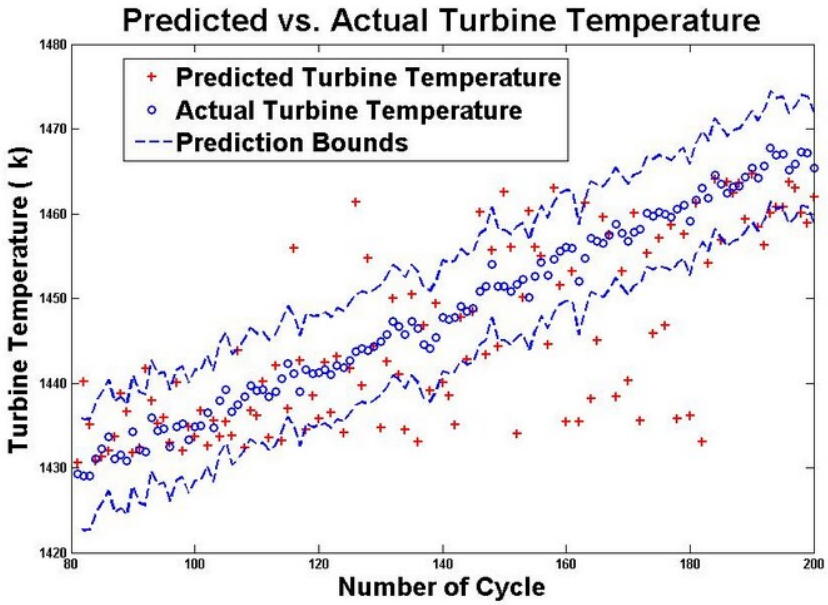

Figure 3.89: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$.

If the training data increases to 120 data points and the same procedure is repeated, the network with 12 hidden neurons has the lowest RMSE as presented in Table 3.81. Comparing Tables 3.80 and 3.81, the RMSE decreases 18.47% by increasing the training data 33.33%. Prediction bounds are depicted in Figure 3.90 to overcome the problem of uncertainty in measurements. In this case, 48.1% of the predicted data are within the lower and the upper prediction intervals for the network structure 7-12-1.

Table 3.82 summarizes the results of the prediction error when 160 data points are used in the training phase and 40 data points are used to evaluate the performance of the networks in 2 flights ahead prediction. When the network has 9 hidden neurons the prediction mean error, standard deviation and RMSE are $2.3837K$, $2.3512K$, and $2.7317K$, respectively. Based on Figure 3.91, 80% of the predicted data are within the upper and the lower prediction bounds for this network.

Next the 5 flights ahead turbine temperatures are also predicted in presence of

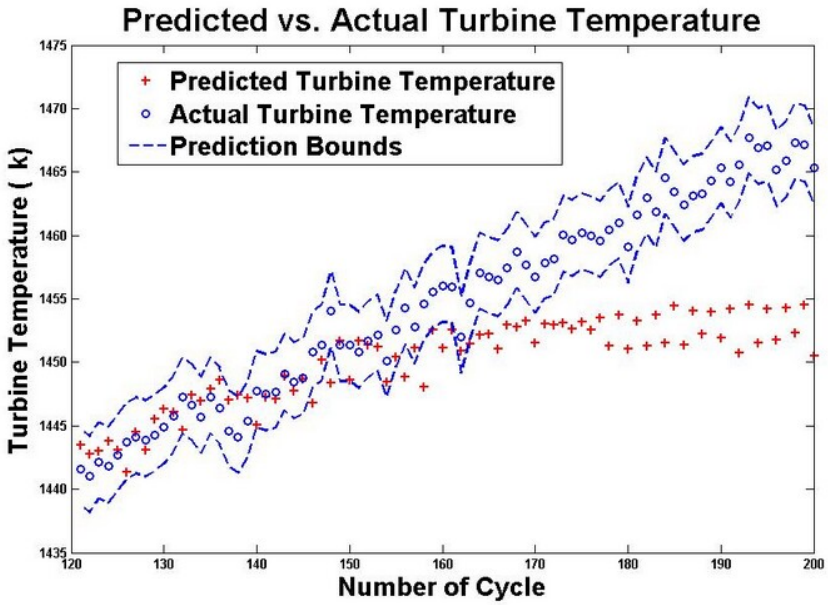

Figure 3.90: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-12-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$.

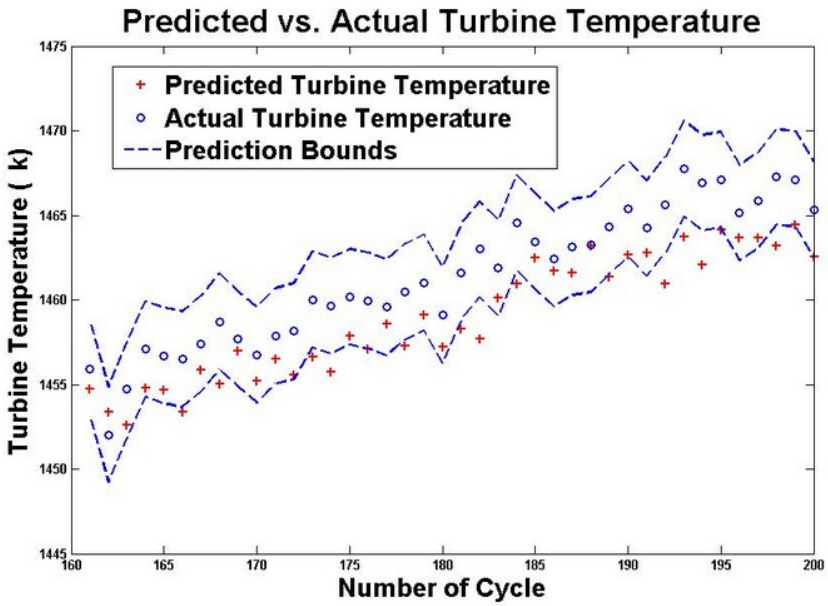

Figure 3.91: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-9-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$.

Table 3.81: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	11.7929	6.5602	13.4749
6	10.0714	8.3660	11.3959
7	8.1696	7.2408	10.8865
8	9.2846	6.7820	10.9186
9	8.4268	6.1911	9.8804
10	5.8724	7.3508	9.3725
11	7.6846	5.6894	8.9872
12	**4.7137**	**5.1611**	**6.9659**
13	8.1895	6.9997	9.5787
14	8.0755	7.5133	10.3492
15	8.6271	8.2438	10.6266

3% compressor fouling and 3% turbine erosion. In the first case, the optimal NARX neural network structure is found by using 80 data points in the training phase and 120 data points in the testing phase. The results are tabulated in Table 3.83 where the network structure with 12 hidden neurons has the lowest RMSE ($12.6999K$). In the second case, the training data increases to 120 data points, while 80 data points are used to test the performance of the networks in 5 flights ahead turbine output temperature prediction. The results for prediction error, standard deviation and RMSE for these networks are summarized in Table 3.84, and finally different NARX neural network structures are trained with 160 data points. The weights and biases stay fixed and 40 unseen data points are given to the networks to predict 5 flights ahead turbine temperature. The results are shown in Table 3.85 where the network with 11 hidden neurons has the lowest RMSE.

Prediction bounds are depicted to overcome the problem of uncertainty in measurements along with the actual and predicted turbine temperatures in Figures 3.92-3.94 for the three cases mentioned previously where 33.33% of the predicted data

Table 3.82: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	5.1657	2.8461	5.8806
6	4.3903	2.8360	5.2073
7	4.2897	2.8835	5.1487
8	2.8039	2.3484	3.6385
9	**2.3837**	**2.3512**	**2.7317**
10	3.3022	1.8746	3.7856
11	1.4828	3.6656	3.9115
12	3.2289	2.6348	4.1466
13	3.5025	2.2159	4.1297
14	4.2605	2.1932	4.7793
15	4.6362	2.7366	5.3662

when the network 7-12-1 is trained with 80 data points are within the prediction bounds. This value increases to 40% for the network trained with 120 data points and 82.5% for the network 7-11-1 which is trained with 160 data points.

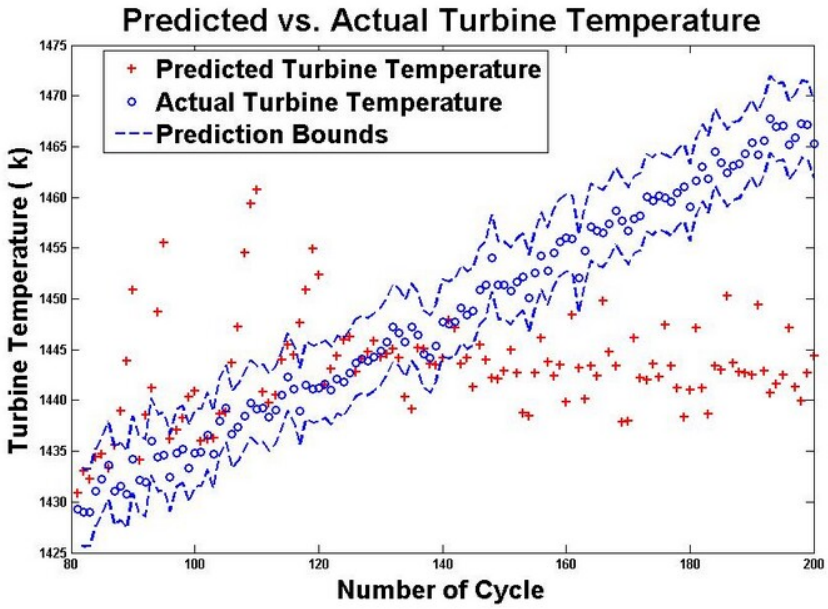

Figure 3.92: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-12-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$.

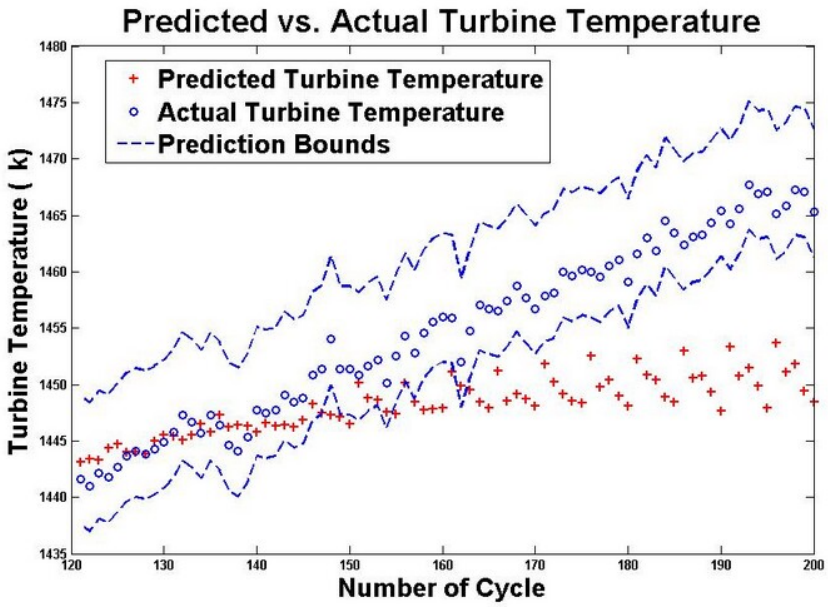

Figure 3.93: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-10-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$.

Table 3.83: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	17.7998	11.5036	21.1675
6	16.5740	8.6363	18.6724
7	13.8457	9.6835	16.8728
8	12.6633	10.0978	16.1702
9	12.8932	8.9331	15.6643
10	11.8321	9.6109	15.2184
11	12.5051	7.5074	14.5695
12	**5.3860**	**11.5495**	**12.6999**
13	11.0525	7.3852	13.2757
14	11.0194	9.3195	14.4069
15	11.7512	10.2836	15.5872

Table 3.84: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	12.7737	7.3885	14.7335
6	12.0195	6.4010	13.5989
7	9.4691	7.1885	11.8614
8	8.6475	6.1071	10.5645
9	5.9013	6.8691	9.0233
10	**6.5175**	**5.9827**	**8.8218**
11	6.8080	7.3540	9.9877
12	8.8265	5.7435	10.5111
13	8.7222	7.0997	11.2184
14	10.4564	7.5270	12.8563
15	10.5410	8.2094	13.3291

Table 3.85: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
5	6.2103	3.5399	7.1264
6	5.9962	3.3085	6.8284
7	5.9843	3.0907	6.7176
8	5.6036	3.0646	6.3684
9	4.9257	3.1214	5.8105
10	3.8496	2.7966	4.7376
11	**2.4658**	**3.0878**	**3.9213**
12	5.1546	3.0497	5.9698
13	5.1422	3.1677	6.0188
14	4.9759	4.6384	6.7629
15	5.6156	5.0267	7.4947

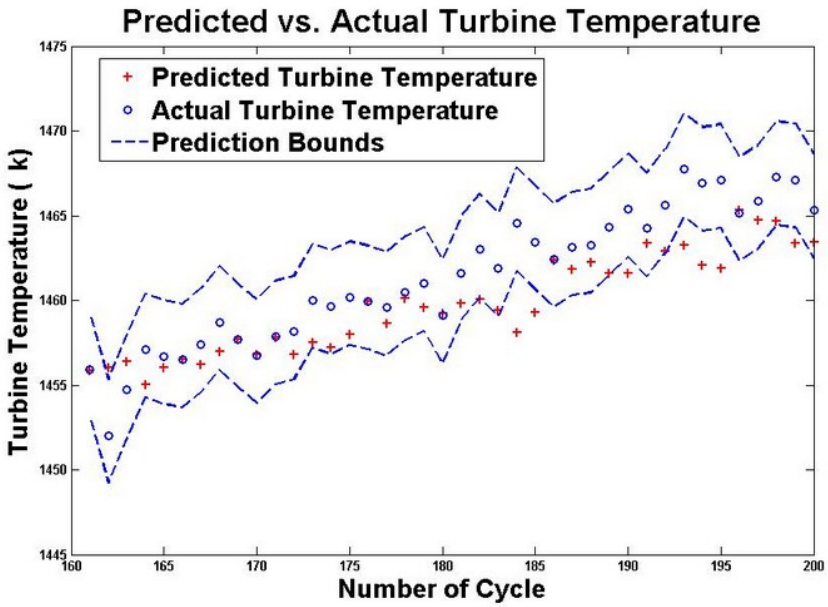

Figure 3.94: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using NARX 7-11-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$.

3.1.3.5 Summary of the Results

Tables 3.86-3.89 show the summary of the RMSE for the optimal NARX neural networks found by trial and error in Section 3.1.3 for different scenarios when the engine has both compressor fouling and turbine erosion at the same time, where N_{train} is the number of data used in the training phase and N_{test} is the number of data which were used to test the trained network, and N_F is the number of flights ahead which the networks are used to predict the turbine temperature.

Table 3.86: Summary of the prediction errors for each scenario in presence of $FI = 1\%$ and $EI = 1\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-9-1	5.8402
120	80	2	7-9-1	4.8784
160	40	2	7-8-1	2.3584
80	120	5	7-11-1	6.2250
120	80	5	7-8-1	5.0015
160	40	5	7-6-1	3.2896
80	120	8	7-10-1	6.4827
120	80	8	7-9-1	5.9867
160	40	8	7-7-1	3.5577
80	120	12	7-11-1	7.9234
120	80	12	7-9-1	6.9783
160	40	12	7-9-1	4.1524

In presence of both fouling and turbine degradations at the same time, the amount of fouling and erosion in the gas turbine engine plays an important role in the network's learning capability. As shown in Table 3.86, the NARX neural network can predict turbine output temperature in 5 flights ahead with the accuracy of 99.76% when it is trained with 80% of the available data points when the turbine has only 1% fouling and 1% erosion where 92.5% of the predicted data points are within the upper and the lower prediction bounds. However, the RMSE increases 16.1% in presence of 3% compressor fouling and 3% turbine erosion. Only 82.5% of the predicted data

Table 3.87: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ and $EI = 2\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-10-1	7.9045
120	80	2	7-9-1	5.7057
160	40	2	7-9-1	2.5643
80	120	5	7-9-1	10.2959
120	80	5	7-10-1	7.5415
160	40	5	7-9-1	3.3064
80	120	8	7-10-1	14.4070
120	80	8	7-11-1	10.5503
160	40	8	7-8-1	4.1737

Table 3.88: Summary of the prediction errors for each scenario in presence of $FI = 2\%$ and $EI = 3\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-8-1	6.6924
120	80	2	7-10-1	4.4616
160	40	2	7-10-1	2.6419
80	120	5	7-11-1	11.5552
120	80	5	7-10-1	6.5188
160	40	5	7-10-1	3.3963
80	120	8	7-12-1	17.2626
120	80	8	7-10-1	9.3160
160	40	8	7-10-1	4.4641

points are within the prediction bounds in this case.

As the number of data which are used to train the network increases, the network prediction error decreases. This is due to the fact that the network can learn the trend of degradation better in presence of more available data. The RMSE decreases 70% when the number of training data increases from 80 data points to 160 data points under $FI = 3\%$ and $EI = 3\%$.

Comparing Tables 3.86 and 3.89, the prediction error increases when the engine

Table 3.89: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ and $EI = 3\%$ using NARX neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	7-11-1	8.5444
120	80	2	7-12-1	6.9659
160	40	2	7-9-1	2.7317
80	120	5	7-12-1	12.6999
120	80	5	7-10-1	8.8218
160	40	5	7-10-1	3.9213

goes through higher degradation rates. The RMSE increases 13.66% when the network predicts 2 flight ahead turbine temperature under presence of 1% compressor fouling and turbine erosion in comparison to presence of 3% degradations.

3.2 Conclusion

In this chapter, a NARX neural network scheme is proposed for turbine output temperature prediction in aircraft jet engines in presence of degradations. The capability of the NARX neural network in multi-flights ahead turbine output temperature prediction has been investigated in detail and a large number of simulation results were presented. Several scenarios for the NARX neural networks were trained where each network corresponds to a specific degradation mode and specific training and testing data sets. The presented simulations demonstrate the effectiveness of the proposed strategy. The NARX neural networks have the potential to capture the dynamics of non-linear systems. It has the ability to learn the trend of the compressor fouling and turbine erosion degradations. In the next chapter we develop Elman network schemes for the turbine temperature prediction in aircraft jet engines to investigate the applicability of this neural network in multi-flights ahead prediction.

Chapter 4

Jet Engine Prediction using Elman Neural Networks

The basic idea behind the condition-based maintenance (CBM) is to make the maintenance strategy based on the condition of the engine. This is considered in this chapter by using the Elman neural networks. The Elman neural networks is applied to the data obtained from the degraded engine in Section 2.6 for learning the trend of the degradation. This chapter is concerned with compressor fouling and turbine erosion. Concurrent degradations may also occur in an engine. It is also assumed that the compressor fouling and turbine erosion take place at the same time. Maximum values of the fuel flow rate and turbine temperature in each flight are stored and these values for multiple flights are used to train the Elman network.

An Elman neural network is a network which in principle is set up as a regular feed-forward network where all neurons in one layer are connected with all the neurons in the next layer. These connections are indicated with weights as described in Section 2.2. The distinction in this network is the context layer which is a special case of a hidden layer. The neurons in this layer hold a copy of the output of the hidden neurons. The values of the context neurons are used as an extra input for all the

neurons in the hidden layer in the time steps later. Using the Elman neural network, the memory is build through feedback from neurons to the first layer. The same set of data sets including the fuel flow rate as an input and the turbine temperature as the output of the network are given to the network several times as the connection weights are refined. The training and weight update laws are described in Section 2.2.1. The weights are adjusted in such a way that the error between the actual output and the network output is reduced. The process stops when the error reaches some statistically desired point. The architecture of the Elman network during training step is depicted in Figure 4.1.

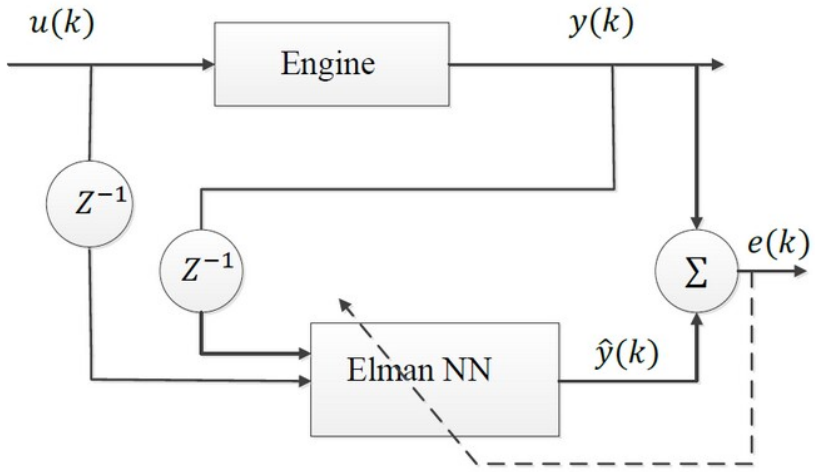

Figure 4.1: An architecture of the Elman network during the training phase.

The trained networks are then used to predict the turbine temperature for various flights ahead. Fuel flow rate is fed to the Elman neural network as an input and the turbine temperature is predicted as the output of the network. Using these predictions, one is able to study the behaviour of key parameters in future to obtain early warnings of possible process malfunctioning. The main problem in time series prediction with neural networks is to find the structure of the network which can represent the dynamics of the system. The schematic view of this network in the

testing phase is shown in Figure 4.2.

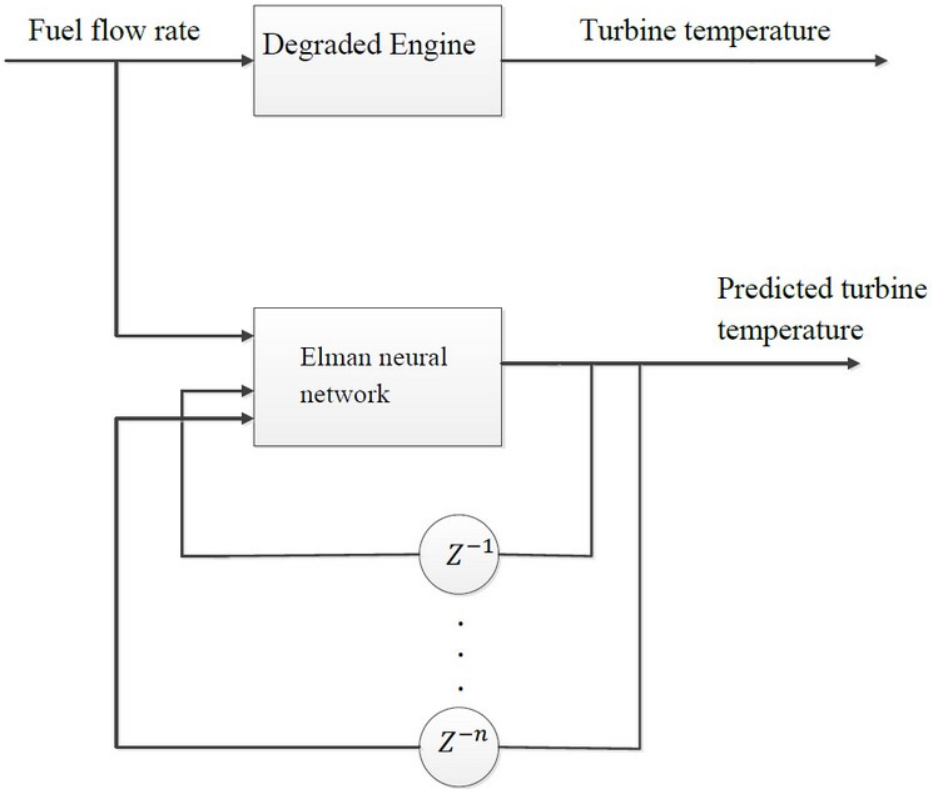

Figure 4.2: A schematic view of the Elman neural network in the testing phase.

4.1 Simulation Results

The performance of the Elman neural network in prediction depends on the number of hidden neurons, the size of the training data set, the number of delays, etc. The number of the data used in the training phase must be sufficient to capture the dynamics of the degradation in the engine, while at the same time the number of hidden neurons must be kept to a minimum in order to minimize the size of the network. For the application under consideration, the fuel flow rate corresponds to the input vectors at a given time step and the target vectors are the turbine output temperature at the following time steps. The number of hidden neurons has to be

adjusted. Optimal Elman neural network structure is achieved by using different number of training data sets to predict 2 flights ahead turbine temperature. We will also use 5 steps, 8 steps and 12 steps ahead turbine temperature to investigate the reliability of the Elman neural network for the long term prediction problem.

4.1.1 Compressor Fouling

One of the important factors which reduces the performance of the gas turbine is the airborne particles entering the engine with the air which are then adhered to the blades of the compressor. This degradation directly affects the compressor performance. As shown in Figure 2.13, the turbine temperature increases in presence of fouling in the compressor which causes a reduction in the turbine output power. Therefore, the fuel flow rate has to be increased to maintain the power in its constant value. This fact emphasizes the importance of predicting the performance of the gas turbine due to the fouling phenomena. It is worth noting that in most engine applications the compressor is washed after 3% of compressor fouling, so it is not useful to predict the turbine temperature for higher fouling indices. Various Elman neural networks are trained and tested in presence of 1% and 3% compressor foulings to predict multi-flights ahead turbine temperature. The predicted values help the operator to base the maintenance decision on the actual deterioration of the system to avoid failures and minimize the cost of maintenance.

4.1.1.1 FI = 1%

The Elman neural network is used in this section to predict the turbine output temperature in presence of 1% compressor fouling. It is assumed that the compressor degrades its efficiency by 1% and its mass flow rate by 0.5% in over 200 flights. The fuel flow rate is given as an input to train the network. The delay is set to 2 which

implies that 2 previous output values of the hidden layer neurons are saved and fed to the network as additional inputs. The number of hidden neurons are increased from 2 to 8 and the 2 flights ahead turbine temperature is predicted. The available data points are 200 where 40% are used to train the network and 120 data are used to evaluate the performance of the network. The results of the prediction error, standard deviation and RMSE are summarized in Table 4.1.

Table 4.1: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.4221	3.7143	6.0584
3	5.0211	2.7122	5.7015
4	**3.0354**	**2.5805**	**4.1086**
5	5.2493	4.7278	5.9105
6	5.9010	4.7382	6.5005
7	6.5658	4.8319	7.1458
8	7.0613	5.7177	7.5621

As shown in Table 4.1, the network with 4 hidden neurons has the lowest RMSE. The predicted data as well as their actual values are depicted pointwise in Figure 4.3. Prediction bounds described in Section 2.4 are also shown to overcome the problem of uncertainty in measurements where only 49.1% of the predicted data are within the upper and the lower bounds. The error which is the absolute difference between the actual and predicted data are shown in Figure 4.4.

Next, the training data is increased to 60% of the entire data set to train the Elman network with different structures. The trained networks are then used to predict 80 unseen data. The statistical errors in prediction are shown in Table 4.2 where the network structure 3-3-1 has the lowest RMSE equal to $2.9178K$. Figure 4.5 shows the actual and the predicted turbine temperatures where 82.5% of the points are within the prediction intervals.

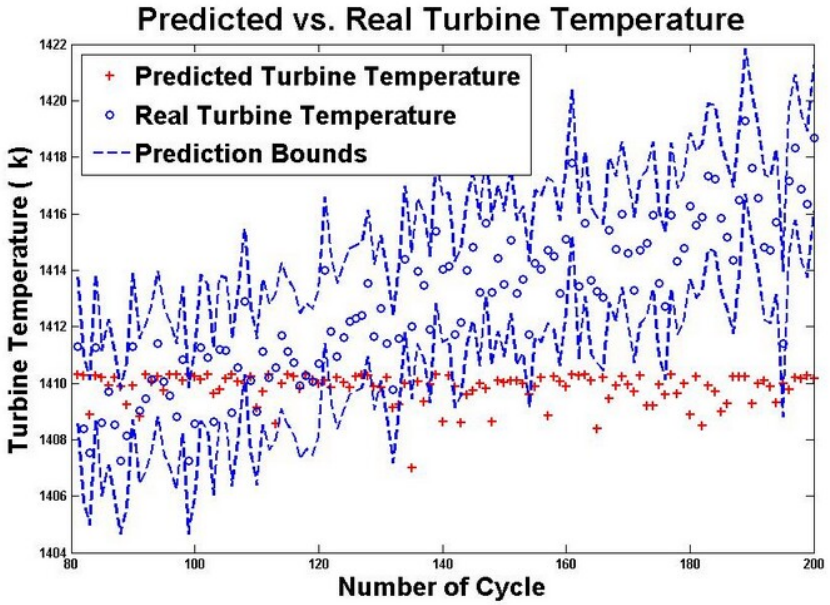

Figure 4.3: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 1\%$.

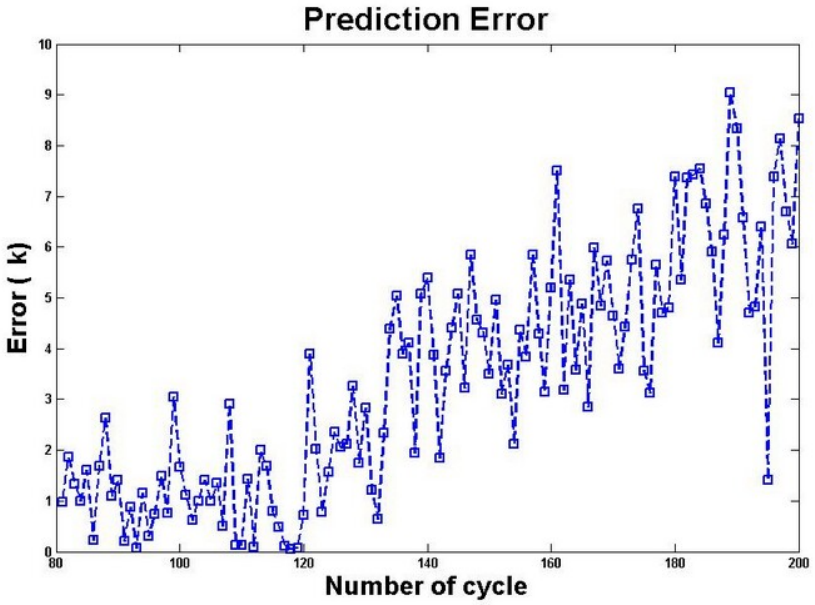

Figure 4.4: Prediction errors for the 2 step ahead turbine temperature when $FI = 1\%$ using the Elman 3-4-1 trained with 40% of the available data.

Table 4.2: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.9574	3.9356	4.4001
3	**1.7575**	**2.3438**	**2.9178**
4	3.6059	2.9223	4.0806
5	4.4195	3.0659	4.8731
6	4.9716	3.0924	5.3889
7	4.7388	4.1915	5.2152
8	6.1164	4.9550	6.4175

In order to investigate the effects of the number of training data sets in the performance of the network, the training data points are increased to 160 points. The number of hidden neurons are increased from 2 to 8 and the Elman networks are tested with the remaining 40 data points. The prediction errors are tabulated in Table 4.3. The predicted values are shown in Figure 4.6 for the network with 3 hidden neurons where 85% of the predicted points are within the prediction bounds.

Table 4.3: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.0331	3.1078	3.6811
3	**1.4535**	**1.8815**	**2.3588**
4	1.8863	1.7135	2.5340
5	2.0669	1.9186	2.8037
6	2.6636	2.0746	3.3603
7	2.7007	2.6699	3.7741
8	2.4821	2.8873	3.7801

Next the 5 flights ahead turbine temperature is predicted by using different number of training and testing data sets. In the first case, 80 data points are used during the training phase and 120 data points are used to evaluate the performance of the

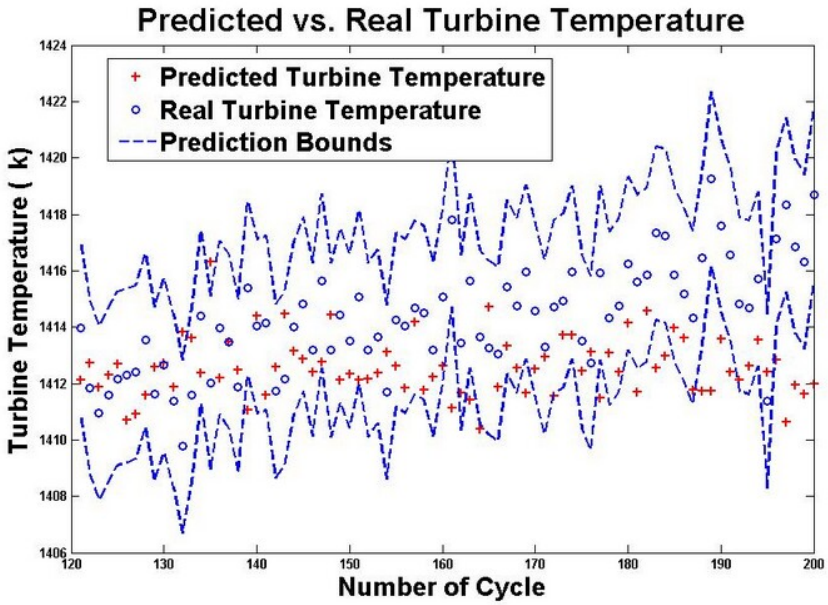

Figure 4.5: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 60% of the available data for $FI = 1\%$.

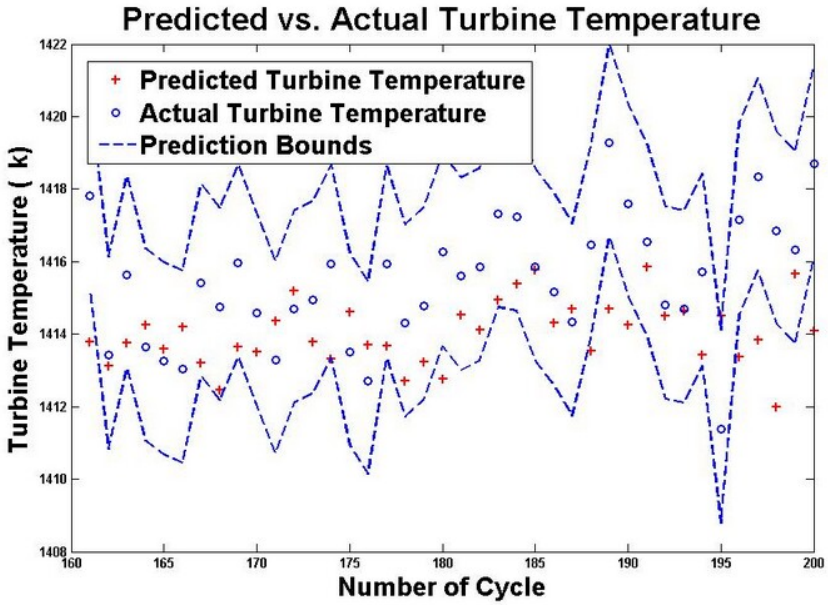

Figure 4.6: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $FI = 1\%$.

networks in prediction. The statistical errors for these network structures are shown in Table 4.4. The network with 5 hidden neurons has the lowest RMSE. The actual and the predicted data along with prediction intervals are depicted in Figure 4.7 where only 46.66% of the predicted data are within the prediction bounds.

Table 4.4: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.3311	3.8182	5.7633
3	4.6236	2.6096	5.3039
4	4.0905	2.7746	4.9362
5	**3.4385**	**2.0460**	**4.3933**
6	3.8670	3.1634	4.9877
7	5.0077	2.6530	5.6619
8	4.9876	2.7641	5.6967

In the second case, the number of training data increases to 120 data points. The number of hidden neurons are changed from 2 to 8 and the Elman networks are trained. These networks are then used to predict 80 unseen data. The performance of these networks in the testing phase are compared together in Table 4.5. The predicted values are shown in Figure 4.8 for the Elman network 3-3-1 where 58.75% of the predicted turbine temperatures are within the prediction intervals.

In the third case, 80% of the entire available data are used to train the Elman networks. The entire data points are 200. Therefore, 160 data points are used in the training phase and the remaining 40 data points are given to the network as inputs in the evaluation phase. The network with 4 hidden neurons has the best performance to predict 5 flights ahead turbine temperature. The mean, standard deviation and RMSE for this network are $1.8568K$, $2.1346K$, and $2.8090K$, respectively as shown in Table 4.6. The temperature difference between the actual and predicted values are depicted in Figure 4.9. Based on Figure 4.10, 85% of the predicted data are within

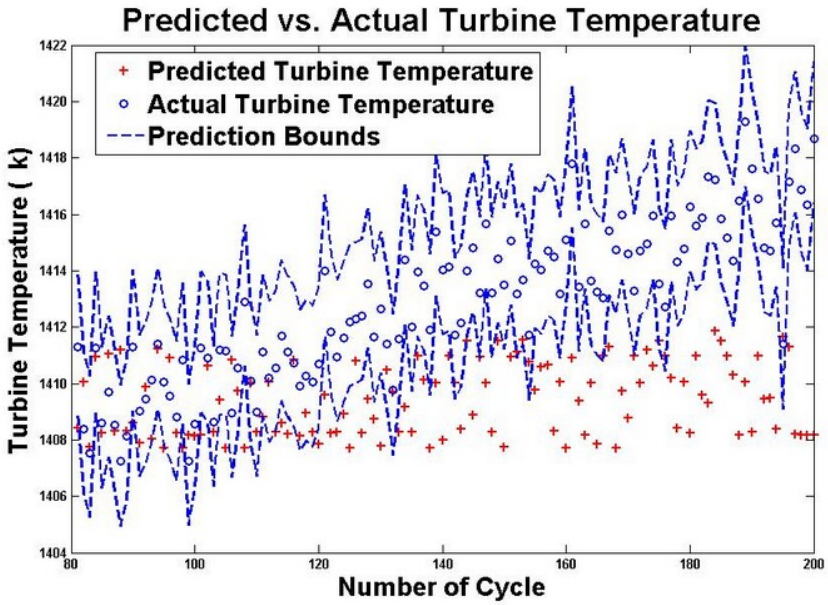

Figure 4.7: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 1\%$.

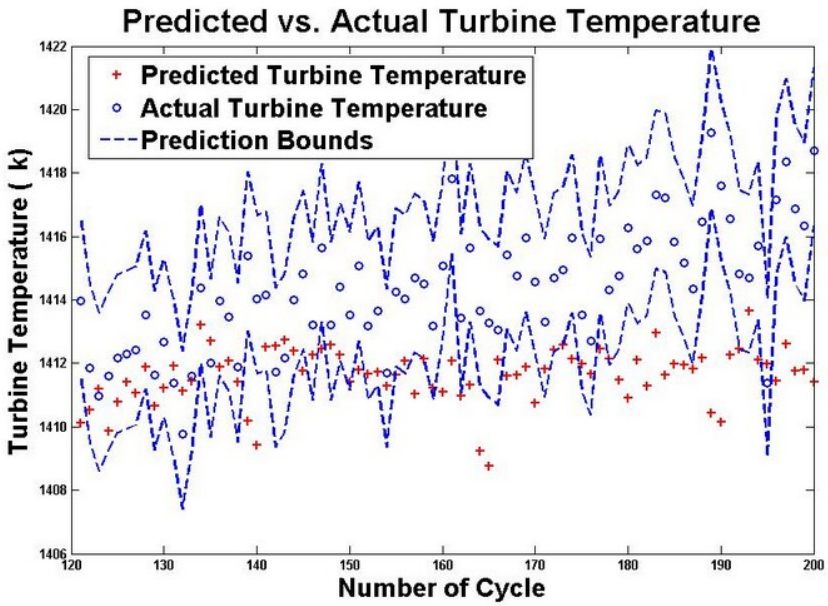

Figure 4.8: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 60% of the available data for $FI = 1\%$.

Table 4.5: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.5879	1.9736	4.0890
3	**2.6970**	**2.0148**	**3.3589**
4	3.0355	2.2890	3.7932
5	3.5579	2.0438	4.0968
6	3.5871	2.2215	4.2119
7	4.1160	3.0452	4.5904
8	4.3405	3.0609	4.7994

the upper and the lower prediction bounds.

Table 4.6: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.9058	2.8100	3.4114
3	2.2774	2.1732	3.1291
4	**1.8568**	**2.1346**	**2.8090**
5	2.3934	2.3405	3.3271
6	0.3854	3.9911	3.9597
7	3.8650	2.2189	4.4428
8	1.5013	4.8120	4.9830

The Elman networks are trained with the available data sets to predict the 8 flights ahead turbine output temperature. The available data sets are divided into 2 for training and testing purposes. First, 80 data points are used in the training phase and 120 points are used to test the networks. The summary of the statistical errors are shown in Table 4.7. Next, 120 data points are used to train the networks and 80 data points in the evaluation phase as shown in Table 4.8, and finally the results of the prediction error for the networks which are trained with 160 data points and tested with 40 data are tabulated in Table 4.9.

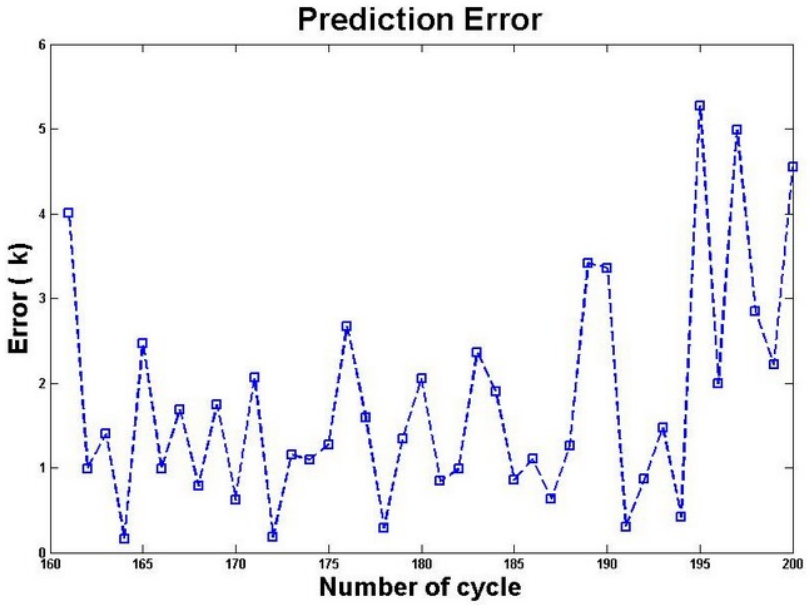

Figure 4.9: Prediction errors for the 5 step ahead turbine temperature when $FI = 1\%$ using the Elman 3-4-1 trained with 80% of the available data.

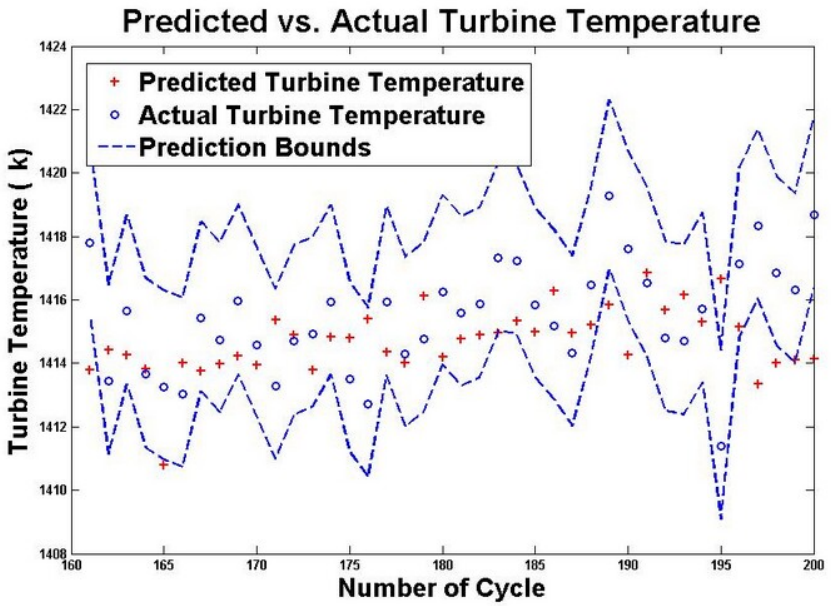

Figure 4.10: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $FI = 1\%$.

Table 4.7: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.7863	3.2036	5.7520
3	4.7890	2.5943	5.4414
4	**4.5621**	**2.8880**	**5.3929**
5	4.8792	3.7455	5.5930
6	4.5664	4.2985	5.6250
7	4.7352	3.2296	5.7241
8	5.3568	4.0432	6.1546

Table 4.8: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.1187	2.5195	5.6982
3	4.9068	3.6793	5.5826
4	3.6651	3.3379	4.9432
5	**4.2592**	**2.1083**	**4.7913**
6	4.7585	3.3668	5.3080
7	5.1974	3.0866	5.5957
8	5.0375	2.2939	5.5292

The predicted data points for the network structure 3-4-1 trained with 80 data points are depicted in Figure 4.11 where only 29.17% of the points are between the upper and the lower prediction intervals. This value increases to 51.25% for the network 3-5-1 which is trained by using 120 data points as shown in Figure 4.12, and 82.5% for the Elman network 3-4-1 trained with 160 data points as can be seen in Figure 4.13.

4.1.1.2 FI = 3%

In this section, the effects of 3% compressor fouling on the turbine temperature is investigated. It is assumed that the engine goes through 3% fouling in 200 flights.

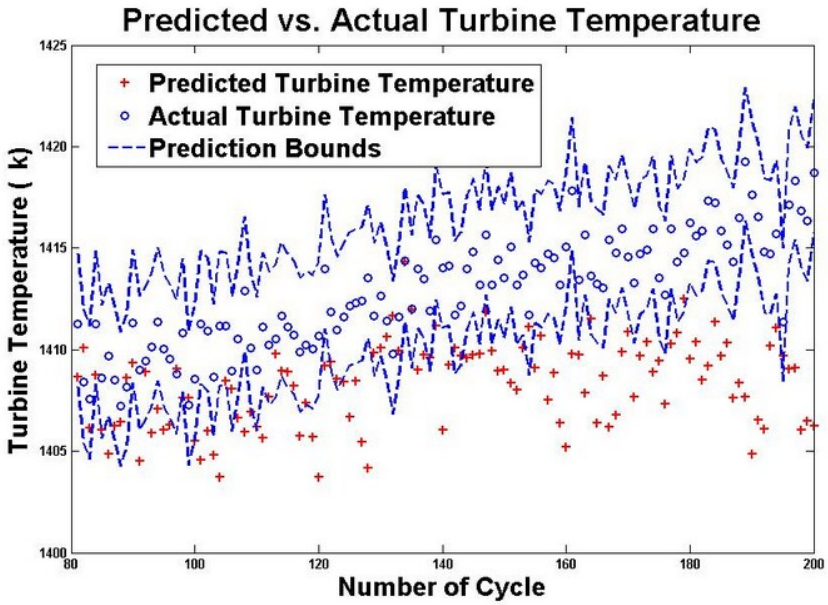

Figure 4.11: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 1\%$.

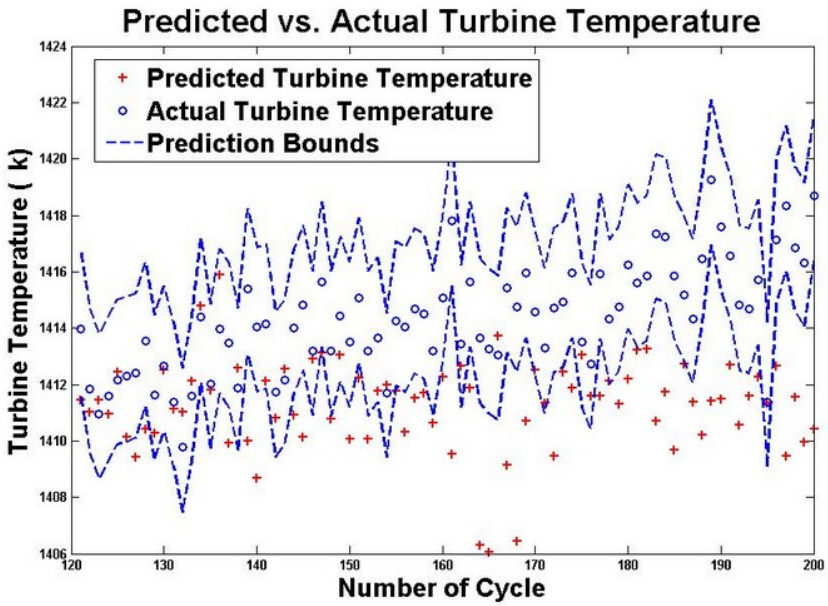

Figure 4.12: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 1\%$.

Table 4.9: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.7337	4.3442	4.6267
3	3.3514	2.1463	3.9653
4	**2.9788**	**2.2637**	**3.7241**
5	3.7869	2.8763	4.2159
6	3.4745	2.5813	4.3092
7	3.6613	3.1281	4.2215
8	4.0108	3.3203	4.6191

The compressor efficiency decreases by 3% and its mass flow rate decreases by 1.5%. As done previously, the 2 flights ahead turbine temperature is predicted by using different number of training data sets. Different Elman network structures are trained by using 40% of the available data points. The remaining 60% is used to predict the turbine temperature. The prediction errors of these networks in the testing phase are summarized in Table 4.10. The network structure 3-5-1 has the lowest RMSE. The actual and prediction values are depicted pointwise in Figure 4.14 where 69.17% of the predicted data are within the prediction bounds.

Table 4.10: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.1526	7.9250	8.1802
3	0.8793	7.2389	7.2621
4	2.5600	4.5839	5.2336
5	**1.5824**	**4.3047**	**4.9451**
6	4.8657	5.1565	7.0741
7	3.6380	6.4317	7.3660
8	5.6037	5.5785	7.8906

Next the Elman networks are trained by using 120 data points. The weights

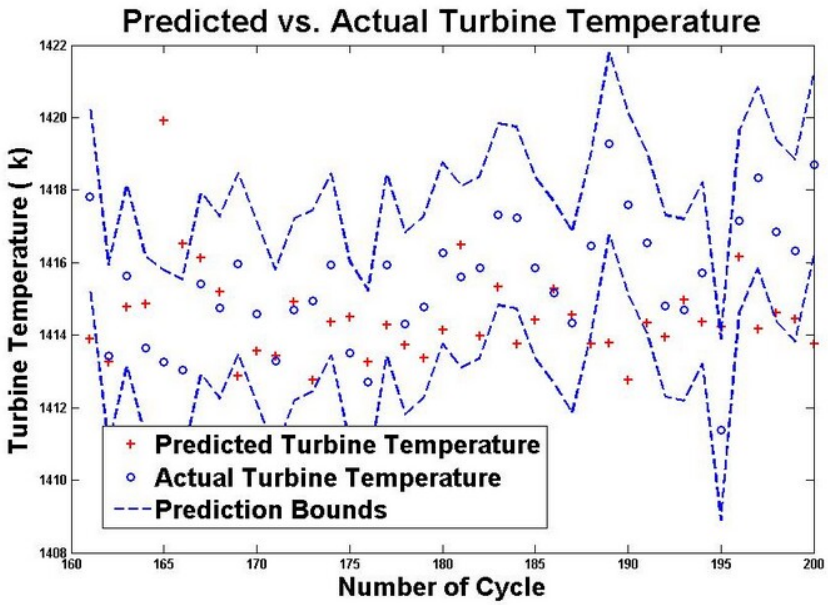

Figure 4.13: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $FI = 1\%$.

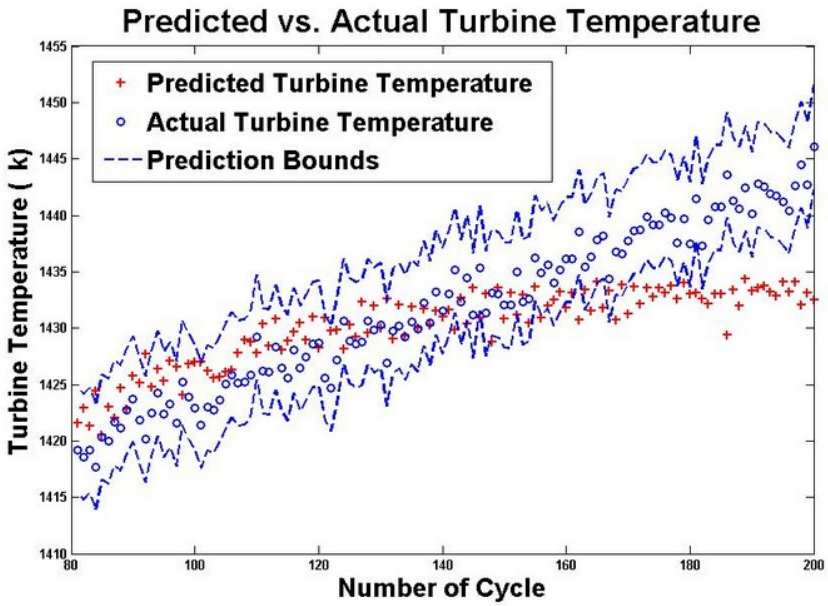

Figure 4.14: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 3\%$.

and biases are fixed and these networks are then evaluated by using 80 data. The error, standard deviation and RMSE are shown in Table 4.11 where the network with 5 hidden neurons has the lowest RMSE equal to $3.8272K$. The predicted data for this network are shown pointwise in Figure 4.15 along with actual data. The prediction bounds are also depicted where 68.75% of the predicted points are within these bounds.

Table 4.11: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.3613	3.5239	5.0311
3	4.1902	2.5762	4.9104
4	3.8160	2.7064	4.6685
5	**2.8965**	**2.5174**	**3.8272**
6	3.5800	2.7312	4.4925
7	3.7093	3.2949	4.9477
8	0.2536	5.6717	5.6418

Training data are increased to 160 data points and the number of hidden neurons are changed from 2 to 8. We used 40 unseen data given to the networks and their applicability in the 2 flights ahead turbine temperature are compared together in Table 4.12. The predicted points for the Elman network with the structure of 3-3-1 are depicted in Figure 4.16 where only 15% of the predicted data are outside the prediction intervals.

Next the 5 flights ahead turbine temperature is predicted using networks with different number of hidden neurons. The entire data set equals to 200 points. A total of 80 data points are used to train the networks and 120 data are used to evaluate the performance of the networks. The statistical measures of errors are tabulated in Table 4.13 where the network with 4 hidden neurons has the lowest RMSE. From Figure 4.17 only 55.83% of the predicted points are within the upper and the lower

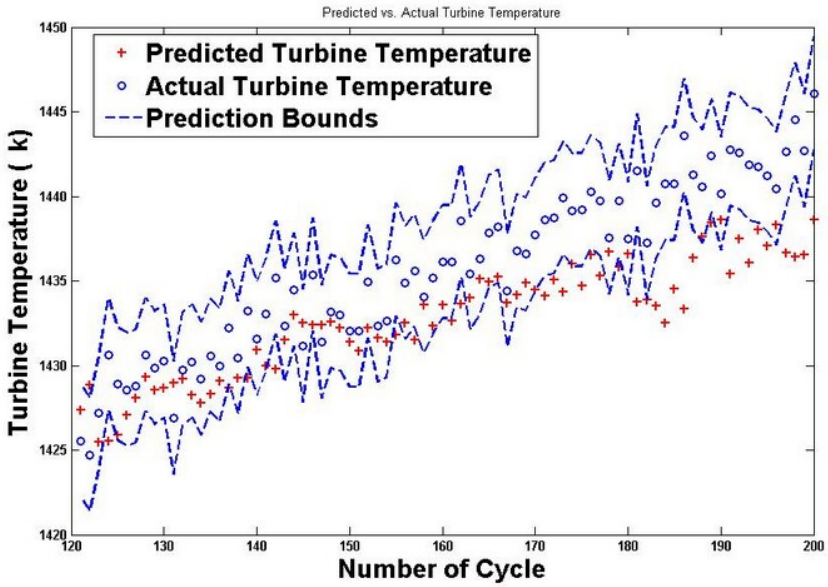

Figure 4.15: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 3\%$.

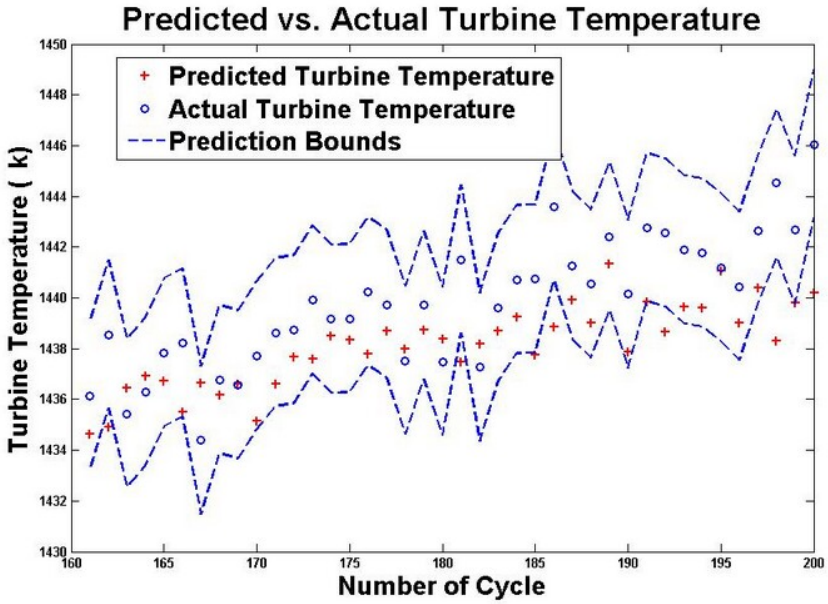

Figure 4.16: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $FI = 3\%$.

Table 4.12: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.2602	2.4097	3.2818
3	**2.2778**	**1.7592**	**2.8646**
4	2.0785	2.5344	3.2531
5	2.9959	1.8021	3.4845
6	3.3084	1.7995	3.7553
7	3.7759	2.2645	4.3883
8	3.9935	2.0857	4.4932

prediction bounds.

Table 4.13: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	6.1215	6.1431	8.6542
3	1.6277	6.9424	7.1024
4	**1.2975**	**5.3967**	**5.5286**
5	4.5535	5.6105	7.2076
6	2.9885	7.0809	7.6585
7	6.4953	6.8698	9.4334
8	8.1291	5.2532	9.6669

Next the training data is increased to 120 data points and the networks are tested by using 80 data points. Based on Table 4.14, the network with 5 hidden neurons has the lowest RMSE equal to $4.3609K$. The actual and predicted turbine temperatures are depicted in Figure 4.18 along with prediction intervals where 55% of the predicted points are within the prediction intervals. The absolute error between the actual and predicted values are also shown in Figure 4.19.

Finally, 80% of the total available data are used to train the Elman networks and the remaining 40 data are used in the evaluating phase. The prediction errors are

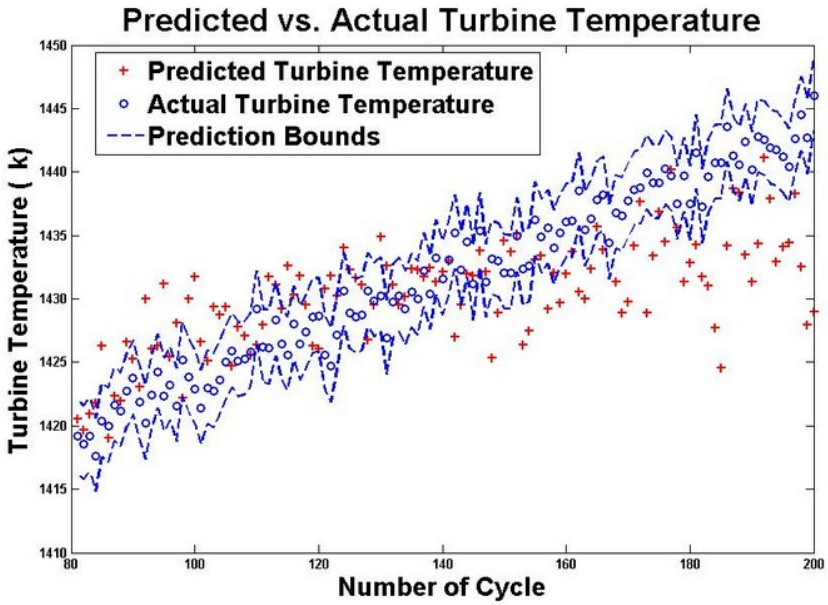

Figure 4.17: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 3\%$.

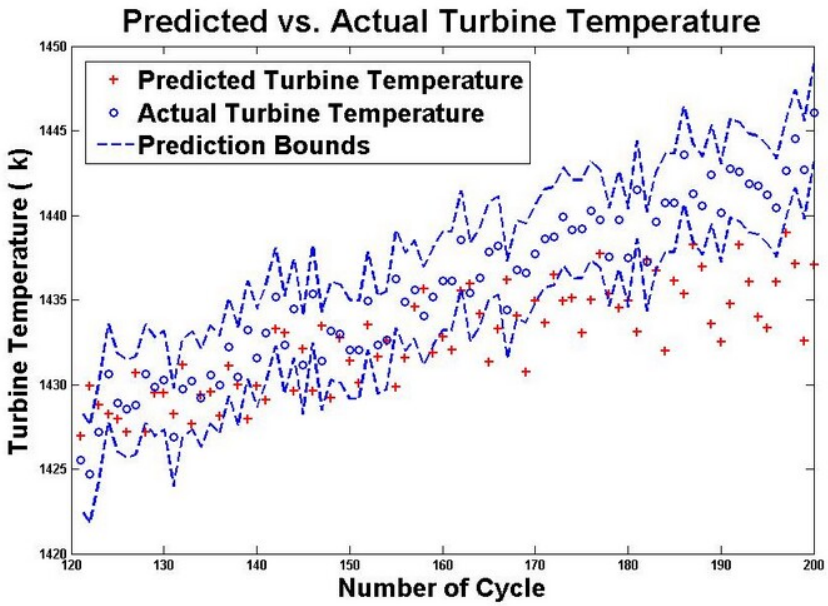

Figure 4.18: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 3\%$.

Table 4.14: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.8020	4.6232	7.4007
3	4.2479	3.7186	5.6303
4	4.3678	3.8873	5.2259
5	**3.0070**	**2.1783**	**4.3609**
6	3.7790	3.6063	5.2081
7	3.4291	4.3768	5.5386
8	0.4297	6.2842	6.2596

summarized in Table 4.15 where the network with 4 hidden neurons has the lowest RMSE. The mean error, standard deviation and RMSE are equal to $2.2243K$, $2.6097K$, and $3.4041K$, respectively. Figure 4.20 shows the predicted turbine temperatures with actual points where 82.5% of the predicted data are within the upper and lower prediction bounds.

Table 4.15: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.6765	2.8520	4.6311
3	2.4686	2.9625	3.8277
4	**2.2243**	**2.6097**	**3.4041**
5	2.5086	2.8682	3.7833
6	3.3746	2.3078	4.0719
7	3.8629	1.8189	4.2600
8	4.0228	2.3693	4.6536

The 8 flights ahead turbine temperature is also predicted in presence of 3% compressor fouling. Three different data sets are used to train the Elman networks. The performance of these networks are then evaluated by predicting the turbine temperature in 8 flights ahead. Table 4.16 is the summary of the prediction errors when the

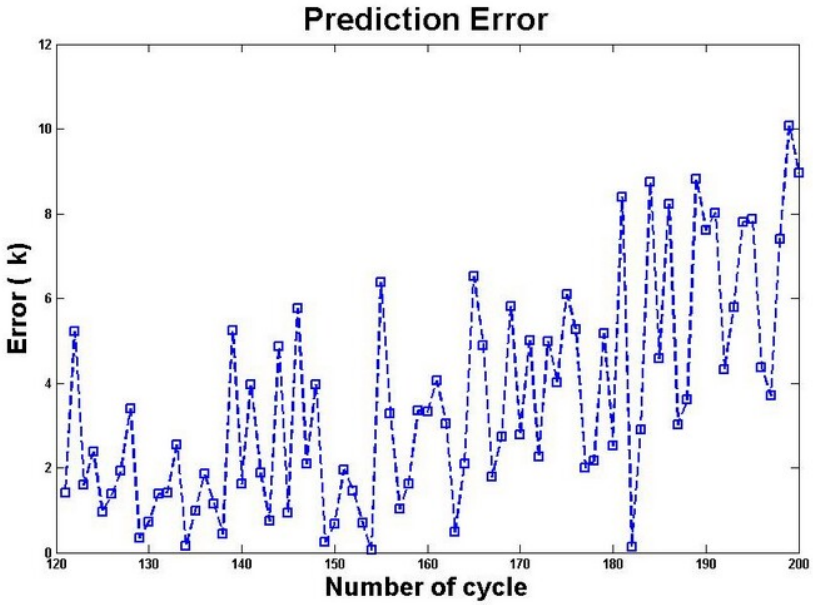

Figure 4.19: Prediction errors for the 5 step ahead turbine temperature when $FI = 3\%$ using the Elman 3-5-1 trained with 60% of the available data.

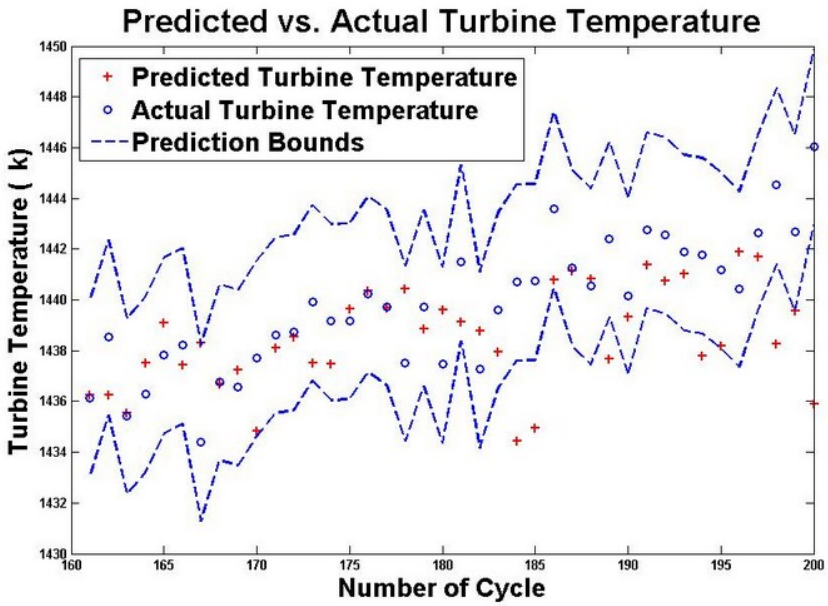

Figure 4.20: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $FI = 3\%$.

networks are trained by using 80 data points and evaluated by using 120 data. When the training data increased to 120 data points the best performance is achieved in the network with 5 hidden neurons as shown in Table 4.17, and finally the RMSE decreases to $4.1986K$ if the Elman network 3-5-1 is trained by using 160 data points as tabulated in Table 4.18.

Table 4.16: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	7.8886	7.9799	11.1973
3	7.6722	6.6721	10.1494
4	**5.5862**	**7.7717**	**9.5447**
5	7.6904	6.5868	10.1077
6	5.4812	9.2997	10.7614
7	6.3179	8.4534	10.5252
8	8.3741	7.4768	11.2055

Table 4.17: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	7.0285	4.8012	8.4949
3	5.6840	5.1847	7.6715
4	6.1822	4.0264	7.3640
5	**2.2337**	**3.7142**	**7.0361**
6	5.9878	6.9509	7.1602
7	6.7399	6.2020	7.9286
8	4.3588	6.9220	8.1434

The actual and predicted data for the three cases mentioned previously are depicted in Figures 4.21-4.23. From Figure 4.21, only 40.83% of the predicted points are within the upper and the lower prediction bounds when the network is trained with 40% of the entire data points. This value increases to 55% if the network is

Table 4.18: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.6993	3.7431	5.4240
3	4.0017	3.8032	5.4878
4	4.1538	2.3348	4.7507
5	**2.9293**	**2.0463**	**4.1986**
6	2.6700	3.5995	4.4454
7	4.5158	4.4817	5.1378
8	4.5257	3.0334	5.4271

trained by using 60% of the available data points, and 77.5% in the case where 80% of the data points are used in the training phase.

4.1.1.3 Summary of the Results

The optimal Elman neural network architectures found in Section 4.1.1.1 are summarized in Table 4.19 for different number of training data sets when the engine goes through 1% fouling in 200 flights. Note that N_{train} is the number of training data and N_{test} is the number of data which were used to test the trained network, and N_F is the number of flights ahead which the networks are used to predict their turbine temperature. The RMSE decreases when the number of training data points increases. This value decreases by 36.06% when the training data points increase from 80 to 160 in predicting 5 flights ahead turbine temperature. Only 46.66% of the predicted points are within the upper and the lower prediction bounds in the first case. However, this value increases to 85% in the second case.

The summary of the optimal Elman networks in presence of 3% fouling in the compressor are illustrated in Table 4.20 where N_{train} is the number of data used in the training phase and N_{test} is the number of data which were used to test the trained network, and finally N_F is the number of flights ahead. The statistical error measures

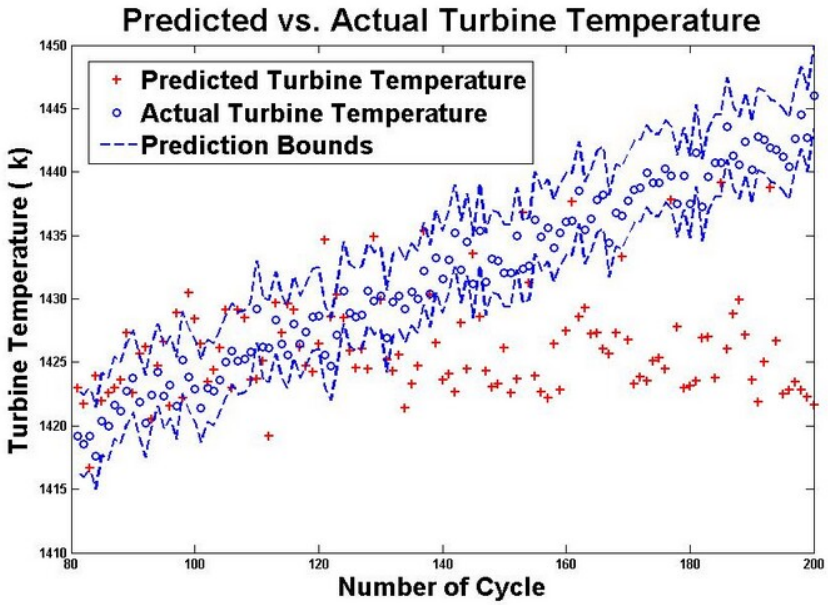

Figure 4.21: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 3\%$.

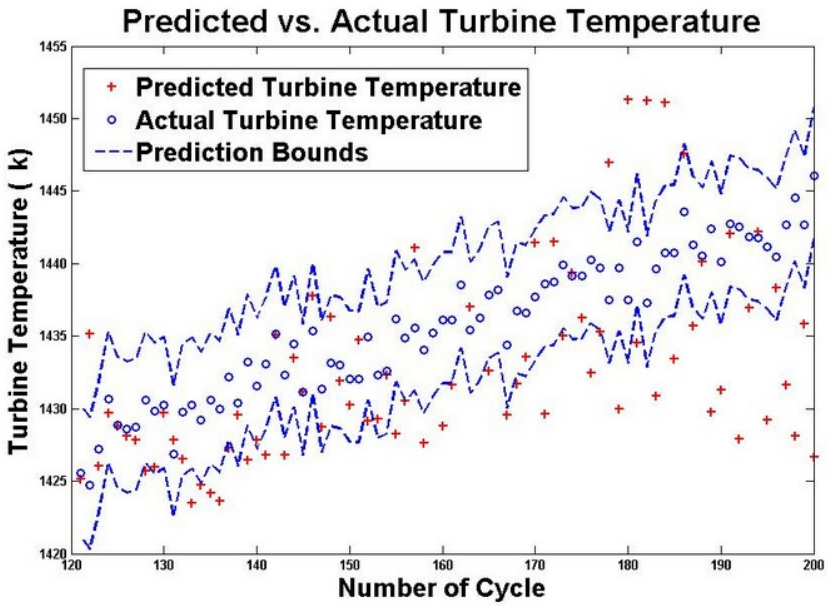

Figure 4.22: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 3\%$.

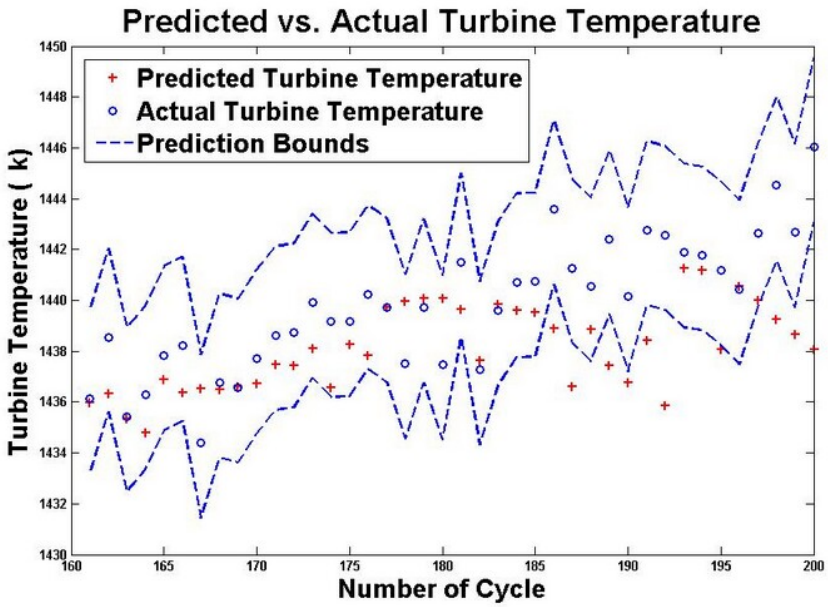

Figure 4.23: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 3\%$.

increase as the number of flights ahead increases. The RMSE of the network used to predict 2 flights ahead is equal to $2.8646K$. This value increases to $4.1986K$ when the Elman network used to predict 8 flights ahead with the same number of training data points. Moreover, as the number of data used in the training phase increases, the network learns the dynamics of the degradation better. Therefore, the error in the prediction decreases. Comparing Tables 4.19 and 4.20, the capability of the Elman network decreases in higher compressor fouling indices.

4.1.2 Turbine Erosion

Erosion in the turbine section of a gas turbine engine is among the main degradations as described in Section 2.6.4 especially for aero engine applications. This deterioration changes the surface blades which affects the blade aerodynamics. Erosion reduces the turbine efficiency and increases the mass flow rate. The effect of erosion is investigated

Table 4.19: Summary of the prediction errors for each scenario in presence of $FI = 1\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-4-1	4.1086
120	80	2	3-3-1	2.9178
160	40	2	3-3-1	2.3588
80	120	5	3-5-1	4.3933
120	80	5	3-3-1	3.3589
160	40	5	3-4-1	2.8090
80	120	8	3-4-1	5.3929
120	80	8	3-5-1	4.7913
160	40	8	3-4-1	3.7241

Table 4.20: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-5-1	4.9451
120	80	2	3-5-1	3.8272
160	40	2	3-3-1	2.8646
80	120	5	3-4-1	5.5286
120	80	5	3-5-1	4.3609
160	40	5	3-4-1	3.4041
80	120	8	3-4-1	9.5447
120	80	8	3-5-1	7.0361
160	40	8	3-5-1	4.1986

on turbine output temperature in presence of 1% and 3% erosion indices. It is assumed that the engine goes under these 2 erosion indices during 200 flights. The data are generated as mentioned in Section 2.5 using equations (2.6.3)-(2.6.4). These degraded data are used to train and test various Elman neural networks to predict turbine output temperature for multi-flights ahead.

4.1.2.1 EI = 1%

In this section the turbine output temperature is predicted under presence of 1% turbine erosion where the efficiency decreases 1% and the mass flow rate increases 0.5% due to removal of the materials from the flow path. The entire data sets are equal to 200 and the delays associated with the hidden layer neurons are set to 2. Also, 40% of the available data points are used to train Elman networks with different structures. The performance of these networks are then evaluated by using 120 data to predict 2 flights ahead turbine temperature. The prediction errors are shown in Table 4.21. The actual and predicted values along with the upper and the lower prediction bounds are depicted in Figure 4.24 for the Elman network with 4 hidden neurons where 91.66% of the predicted data are within the bounds.

Table 4.21: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.2051	2.3220	3.9521
3	2.7708	1.7813	3.2899
4	**1.4919**	**2.0725**	**2.5466**
5	2.2927	2.0167	3.0479
6	2.7571	1.9800	3.3896
7	2.6015	2.6715	3.7209
8	2.9971	2.7785	4.0790

Next the training data points increase to 120 and the number of hidden neurons are increased from 2 to 8 to find the optimal Elman network structure to predict 2 flights ahead turbine temperature. The mean of the prediction error, standard deviation and RMSE for these networks in the testing phase are shown in Table 4.22. The predicted points for the network 3-3-1 are depicted pointwise in Figure 4.25 where 100% of these values are within the prediction intervals.

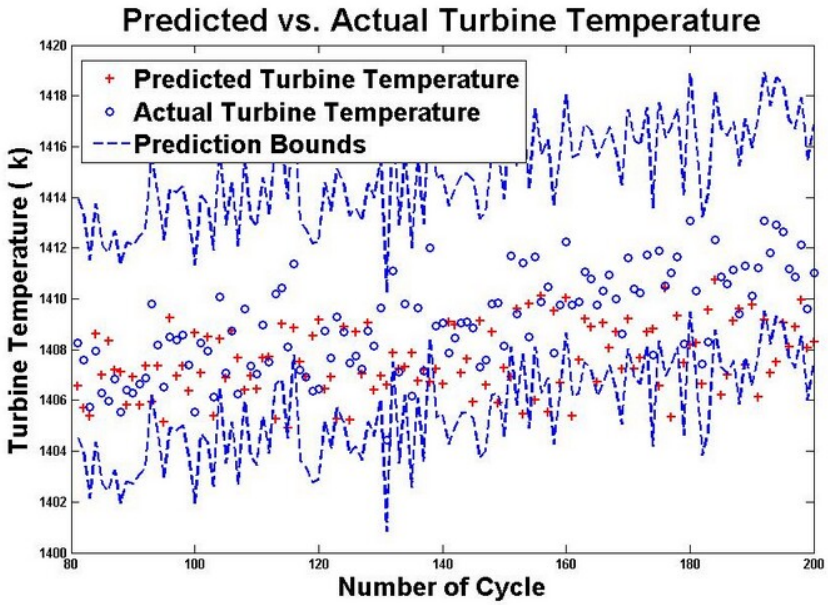

Figure 4.24: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $EI = 1\%$.

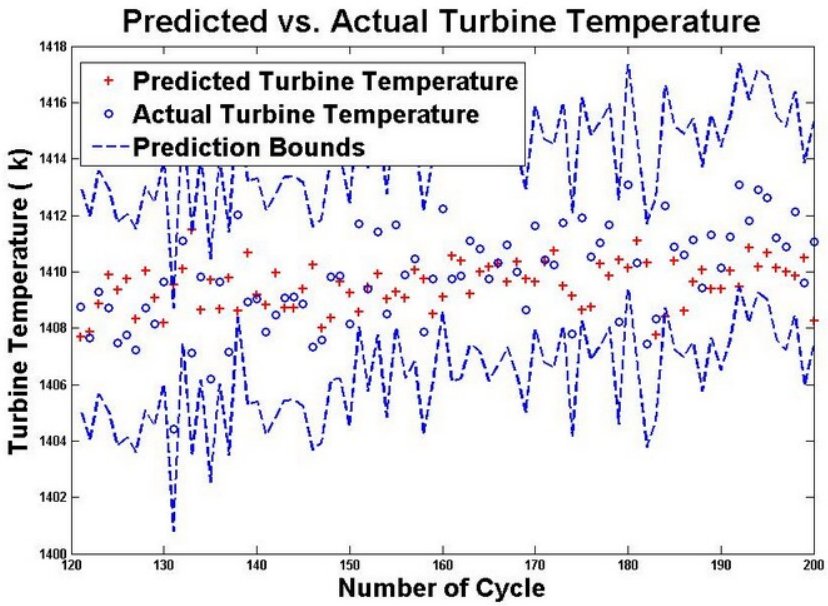

Figure 4.25: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 60% of the available data for $EI = 1\%$.

Table 4.22: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.5922	2.0307	2.5704
3	**0.2939**	**1.8504**	**1.8621**
4	1.1517	2.1274	2.4075
5	2.1017	1.9103	2.8320
6	2.5500	1.7753	3.1008
7	1.1878	3.2584	3.4490
8	3.0355	1.9750	3.6147

When the data used in the training phase increase to 80% of the entire data points, the network with 3 hidden neurons has the best performance based on Table 4.23 where the mean, standard deviation, and RMSE of the prediction error are $0.2020K$, $1.4767K$, and $1.4721K$, respectively. Figure 4.26 shows the actual and predicted points for this network where 100% of the networks output temperatures are within the prediction bounds.

Table 4.23: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.7920	1.5929	2.3843
3	**0.2020**	**1.4767**	**1.4721**
4	0.3735	1.9734	1.9840
5	1.4073	1.9311	2.3699
6	2.1787	1.4050	2.5829
7	2.2195	1.3777	2.6033
8	2.2367	1.4476	2.6544

The 5 flights ahead turbine temperature is now predicted as done previously by using different number of training data sets. In the first case, the training data points are 80 and the remaining 120 data are given to the trained network to test

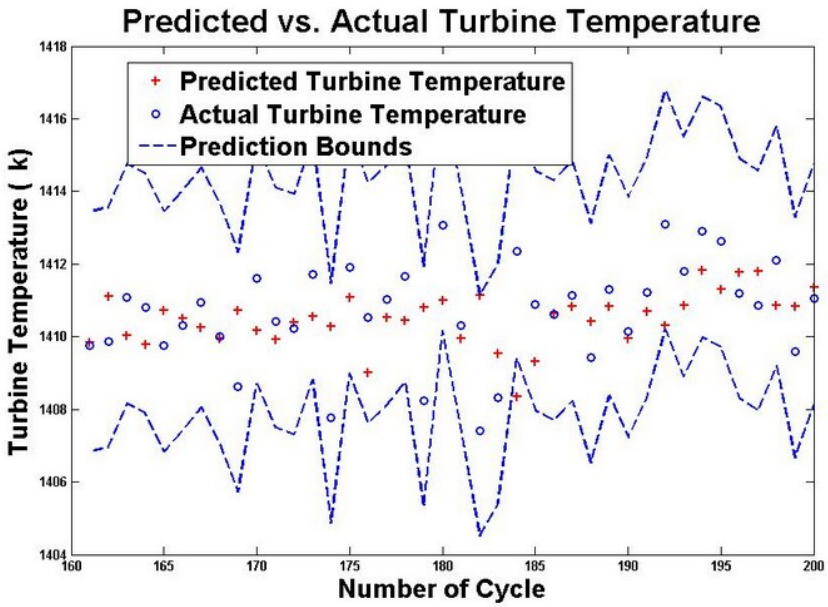

Figure 4.26: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $EI = 1\%$.

the networks. The results of the prediction error are presented in Table 4.24. By increasing the number of training data points, the best Elman network performance is achieved when the network has the structure of 3-4-1 as summarized in Table 4.25. Finally, various networks are trained by using 160 data points. The weights and biases are fixed and 40 unseen data are given as inputs to the networks to predict 5 flights ahead turbine temperatures. The RMSE for the network with 3 hidden neurons is $1.8563K$ as shown in Table 4.26.

Actual and predicted values for the network structure 3-3-1 trained by using 80 data points are depicted in Figure 4.27 where 80.83% of the predicted values are within the upper and the lower prediction bounds. This value increases to 92.5% as shown in Figure 4.28 when 120 data points are used to train the Elman network 3-4-1, and finally 100% when the network with 3 hidden neurons (network architecture 3-3-1) is trained by using 160 data points and evaluated by using 40 unseen data as depicted in Figure 4.29.

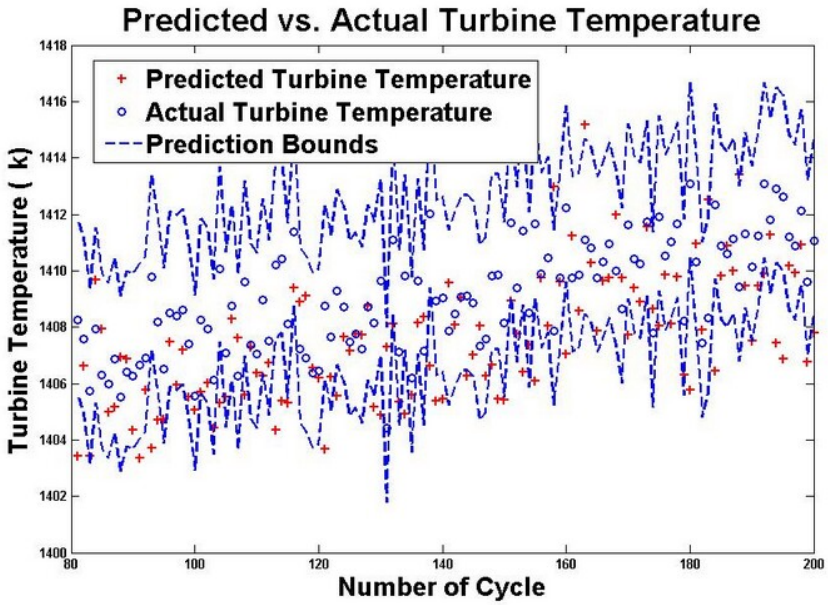

Figure 4.27: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 40% of the available data for $EI = 1\%$.

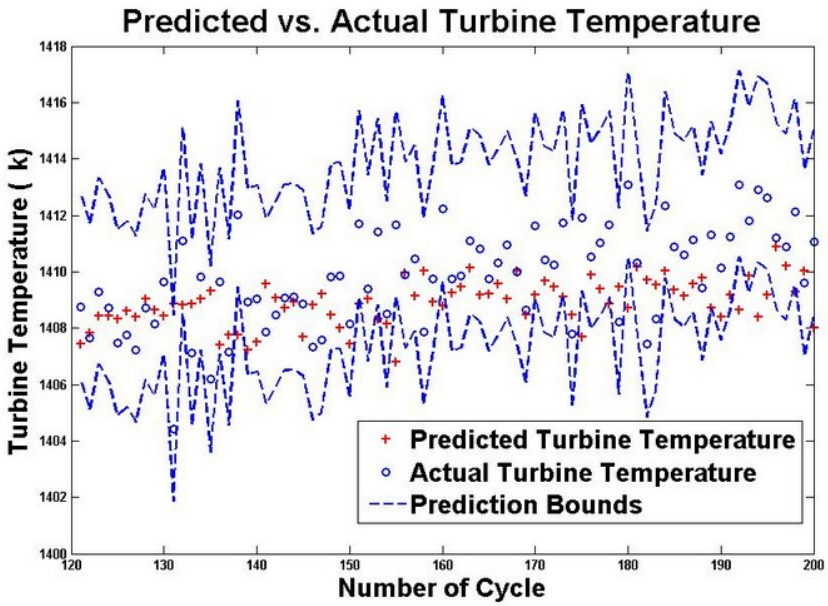

Figure 4.28: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 60% of the available data for $EI = 1\%$.

Table 4.24: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.3515	1.9863	3.0728
3	**1.5365**	**2.3477**	**2.7976**
4	2.1778	2.3382	3.1882
5	1.8850	2.6803	3.2676
6	1.3360	3.4490	3.6853
7	2.2720	2.8467	3.6329
8	3.1514	2.3746	3.9400

Table 4.25: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.7647	2.3627	2.9371
3	2.1105	1.9357	2.8556
4	**0.9143**	**1.8265**	**2.0323**
5	0.6139	2.3604	2.4246
6	2.1108	1.8547	2.8022
7	1.0219	1.8514	2.1045
8	1.9573	2.4834	3.1498

The applicability of the Elman network to predict 8 flights ahead turbine temperature is examined in presence of 1% turbine erosion in over 200 flights. Also 40% of the available data points are used to train the networks with various structures and 60% to evaluate their performance. The summary of the statistical error measures are presented in Table 4.27 where the network with 5 hidden neurons has the best performance. The actual and predicted data are shown in Figure 4.30 where 78.33% of the points are between the upper and the lower prediction bounds.

The effect of the number of training data in the performance of the Elman neural network is investigated by increasing the number of training data to 120 and 160. The number of hidden neurons are changed from 2 to 8, and these networks are trained

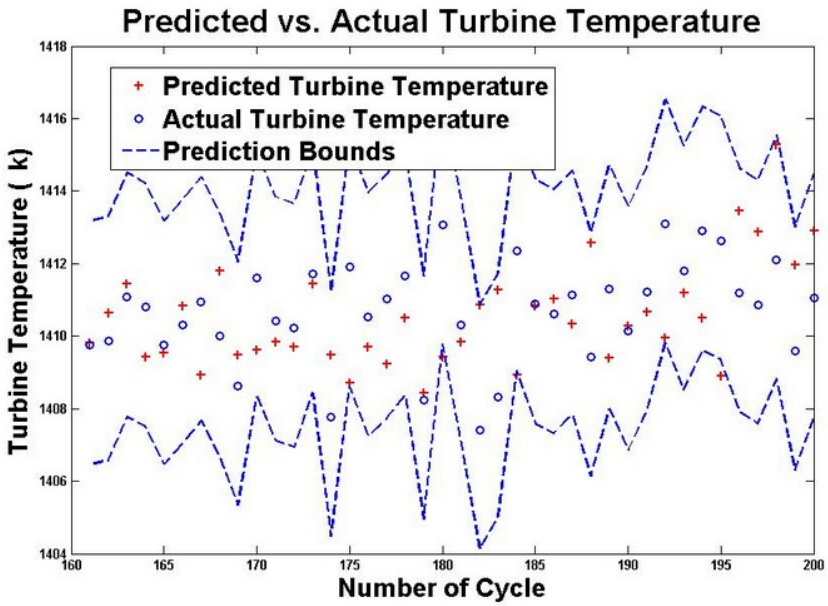

Figure 4.29: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $EI = 1\%$.

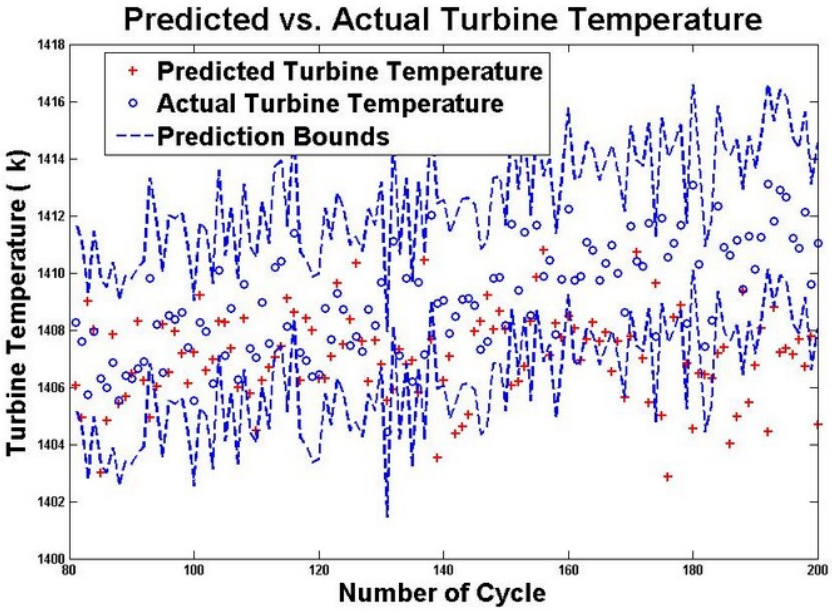

Figure 4.30: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $EI = 1\%$.

Table 4.26: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.7920	1.5929	2.3843
3	**0.1666**	**1.8740**	**1.8563**
4	0.3735	1.9734	1.9840
5	1.4073	1.9311	2.3699
6	2.1787	1.4050	2.5829
7	2.2195	1.3777	2.6033
8	2.2367	1.4476	2.6544

Table 4.27: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.0273	2.8738	4.4386
3	3.6402	2.0393	4.1684
4	3.0594	2.0679	3.6879
5	**2.0314**	**2.4874**	**3.2034**
6	3.3177	2.2158	3.9845
7	3.6050	2.3152	4.2792
8	3.8453	2.2222	4.4366

by using 120 and 160 data points. These trained networks are then tested by using 80 and 40 unseen data to predict 8 flights ahead turbine temperature. The prediction errors are then calculated and summarized in Tables 4.28 and 4.29 for the networks trained by using 120 and 160 data points, respectively.

The actual and predicted values along with the upper and the lower prediction intervals for the network 3-3-1 trained by using 120 data points are shown in Figure 4.31 where 91.25% of the predicted points are within these bounds. This value increases to 97.5% when this network is trained by using 160 data points as depicted in Figure 4.32.

For the next scenario different Elman network structures are trained to predict

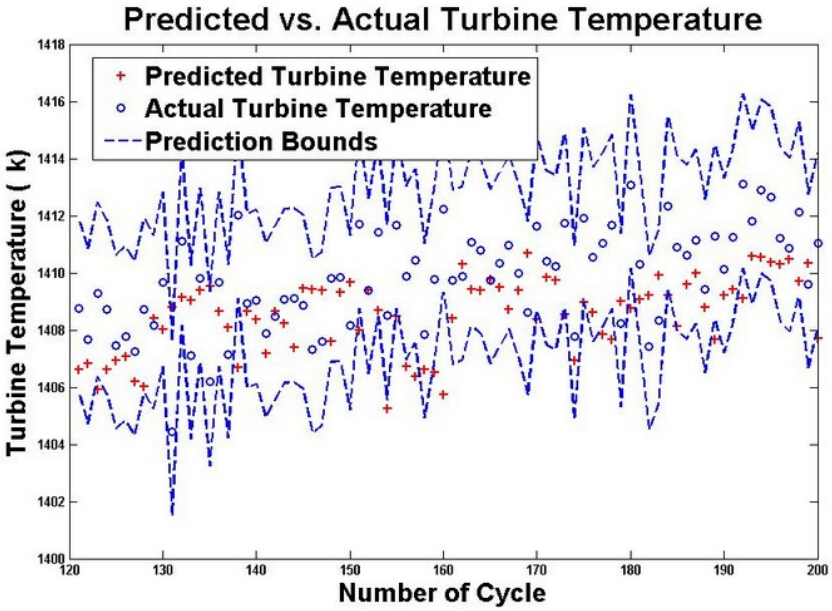

Figure 4.31: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 60% of the available data for $EI = 1\%$.

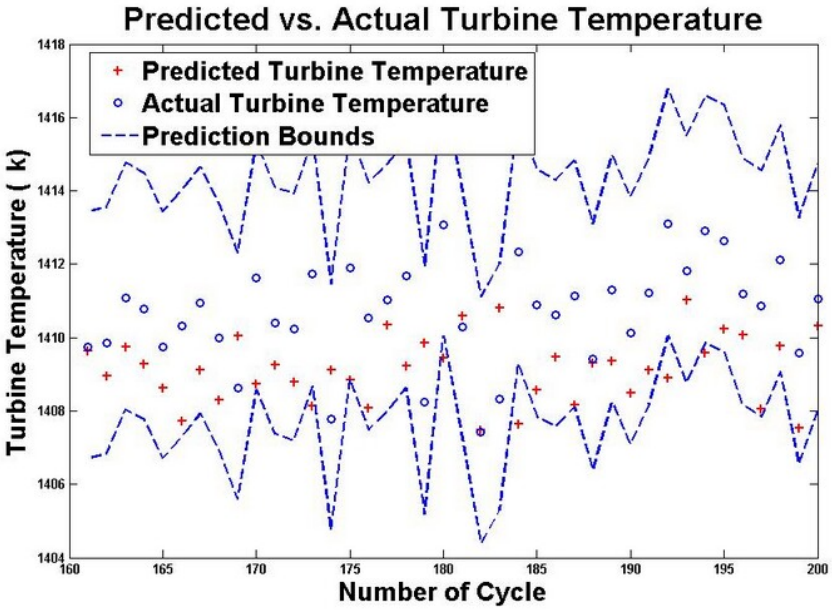

Figure 4.32: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $EI = 1\%$.

Table 4.28: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.0909	2.0631	2.9283
3	**1.2932**	**1.9471**	**2.3273**
4	2.0041	1.6743	2.6047
5	1.6937	2.3141	2.8561
6	2.0932	2.2967	3.0968
7	2.6704	2.0043	3.3314
8	2.3664	2.6062	3.5081

Table 4.29: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.0904	1.6661	2.6601
3	**1.5485**	**1.5899**	**2.2051**
4	1.7094	1.4454	2.4269
5	2.2707	1.3924	2.6545
6	2.6372	1.4030	2.9789
7	0.8705	3.1307	3.2115
8	2.9146	1.4503	3.2474

12 flights ahead turbine temperature. A total number of 80 data points is used in the training phase and the networks are then tested by using 120 data. The network with 4 hidden neurons has the best performance as presented in Table 4.30 where the prediction error, standard deviation and RMSE are $2.4208K$, $2.7697K$, and $3.6699K$, respectively. Based on Figure 4.33, 73.33% of the predicted points are within the prediction bounds.

When the Elman networks are trained with 60% of the entire data points available the lowest RSME is $2.5350K$ for the network with 4 hidden neurons. The prediction errors for various Elman network architectures are tabulated in Table 4.31. To overcome the problem of uncertainty in measurements, the prediction bounds are depicted

Table 4.30: A 12 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.4905	3.1294	4.6793
3	3.8797	2.0227	4.3715
4	**2.4208**	**2.7697**	**3.6699**
5	3.4015	2.0219	3.9528
6	3.9324	2.1531	4.4790
7	4.1881	2.0474	4.6580
8	4.4698	2.1541	4.9578

in Figure 4.34 where 94.44% of the predicted points are within these bounds.

Table 4.31: A 12 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.3826	1.9272	3.0560
3	2.0902	2.1537	2.9905
4	**0.0413**	**2.5524**	**2.5350**
5	1.8062	2.2274	2.8557
6	2.5666	1.9981	3.2441
7	2.6244	1.8415	3.1987
8	2.7896	2.0363	3.4454

Elman networks are next trained by using 160 data points in the training phase and 40 data in the testing phase. The results of the prediction error are compared together in Table 4.32. The actual and predicted values for the network 3-5-1 are shown pointwise in Figure 4.35 where 100% of the predicted values are within the prediction bounds.

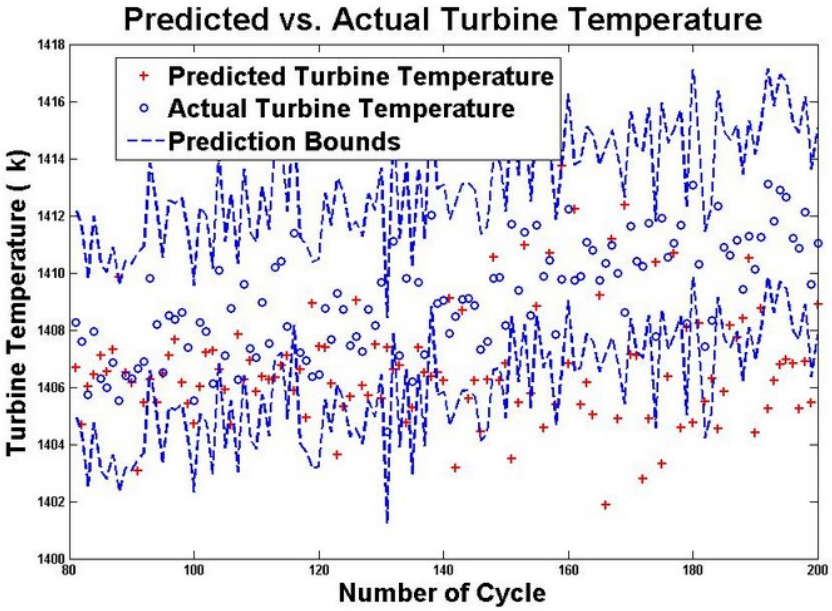

Figure 4.33: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $EI = 1\%$.

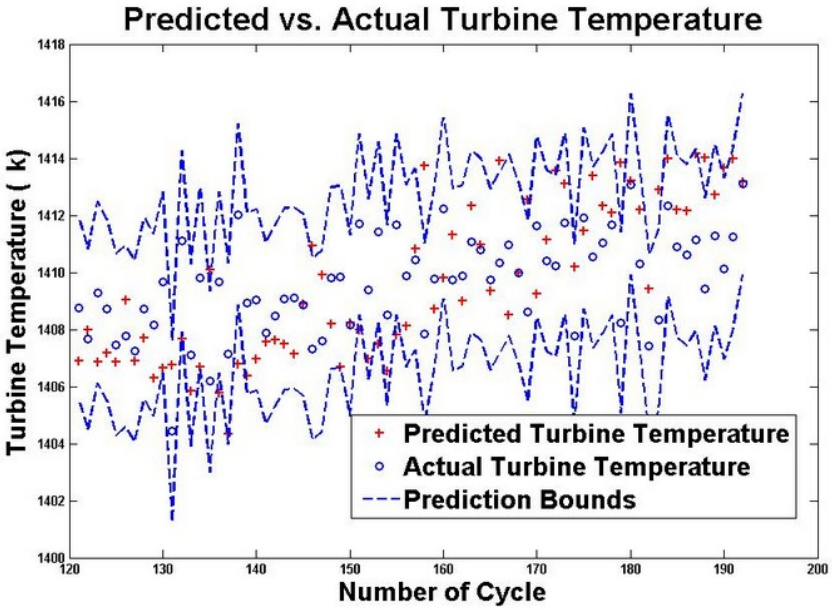

Figure 4.34: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 60% of the available data for $EI = 1\%$.

Table 4.32: A 12 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.7990	2.4255	2.9927
3	2.3489	1.5642	2.8100
4	2.2740	1.5355	2.7319
5	**1.8962**	**1.0982**	**2.4037**
6	2.0285	1.5845	2.5604
7	2.1477	1.6575	2.6988
8	2.3232	1.9666	3.0261

4.1.2.2 EI = 3%

The Elman network is used to predict the turbine temperature in presence of 3% turbine erosion. The turbine efficiency decreases by the amount of 3%, while at the same time the mass flow rate increases by 1.5%. It is considered that the turbine eroded in 200 simultaneous flights. The fuel flow rate is given as an input to train the networks. A total of 80 data points are used to train the networks with different structures. The trained networks are then tested with 120 data points. The prediction errors are tabulated in Table 4.33 where the network with 4 hidden neurons has the best performance. The predicted values are shown pointwise in Figure 4.36 where 85.83% of the predicted points are within the prediction intervals.

Next the training data is increased to 120 data points. The number of hidden neurons are changed from 2 to 8 and 2 flights ahead turbine temperature is predicted for 80 points. The error, standard deviation and RMSE of prediction for these networks are presented in Table 4.34. Actual and predicted data are depicted in Figure 4.37 for the Elman network 3-5-1 where 76.25% of the predicted data are within the prediction bounds. Comparing Tables 4.33 and 4.34 the RMSE decreases by 8.67% when the training data points increases by 33.33%.

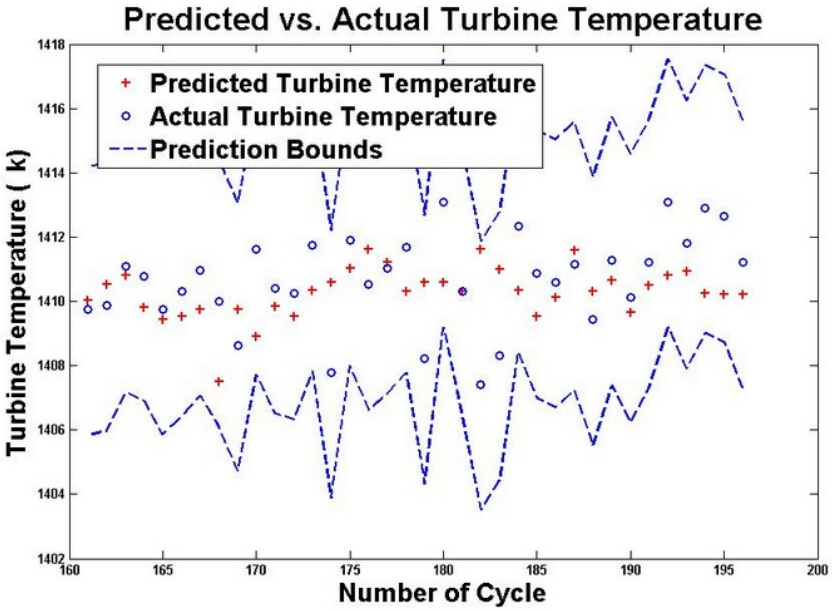

Figure 4.35: The 12 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $EI = 1\%$.

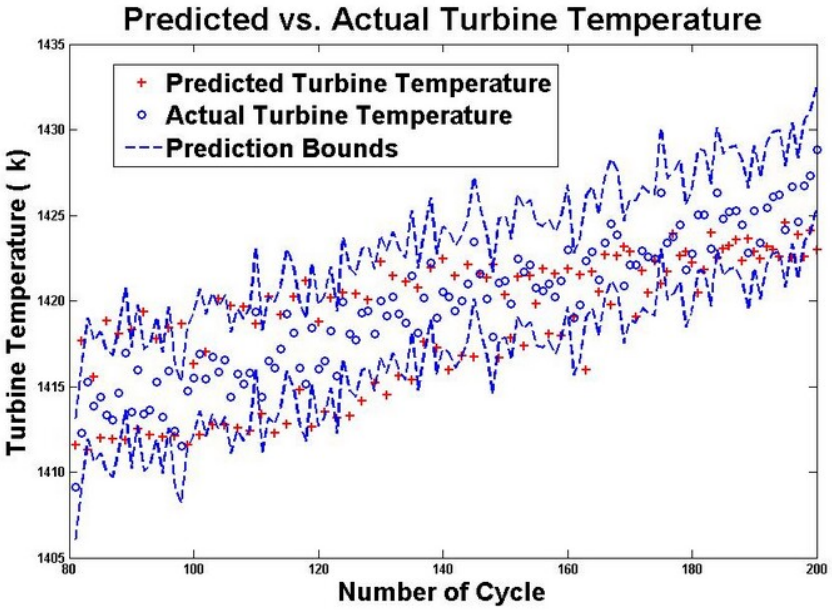

Figure 4.36: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $EI = 3\%$.

Table 4.33: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.5718	3.2873	4.1630
3	1.0577	3.5509	3.6909
4	**0.8590**	**3.1073**	**3.2113**
5	2.4657	3.7638	4.4865
6	2.8403	4.1611	5.0237
7	4.2098	3.0841	5.2111
8	2.5513	4.9191	5.5232

Table 4.34: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.1353	3.3126	4.5460
3	0.8692	4.3252	4.3851
4	2.1267	3.3157	3.9217
5	**2.2403**	**1.9048**	**2.9329**
6	2.4280	2.2548	3.3039
7	3.8887	2.0080	4.3708
8	3.9042	2.4928	4.6238

By increasing the number of training data sets to 80% of the entire available data, the Elman network with 3 hidden neurons has the best performance in predicting the turbine temperature for 2 flights ahead as shown in Table 4.35. Based on Figure 4.38, only 5% of the predicted values are outside the upper and the lower prediction intervals.

In the next scenario, 5 flights ahead turbine temperature is predicted when the engine goes through 3% erosion in the turbine section. First, 80 data points are used to train the network and 120 data points to evaluate the networks performance for various Elman network structures. The summary of the prediction errors are shown in Table 4.36. The network structure 3-4-1 has the best performance. The mean error,

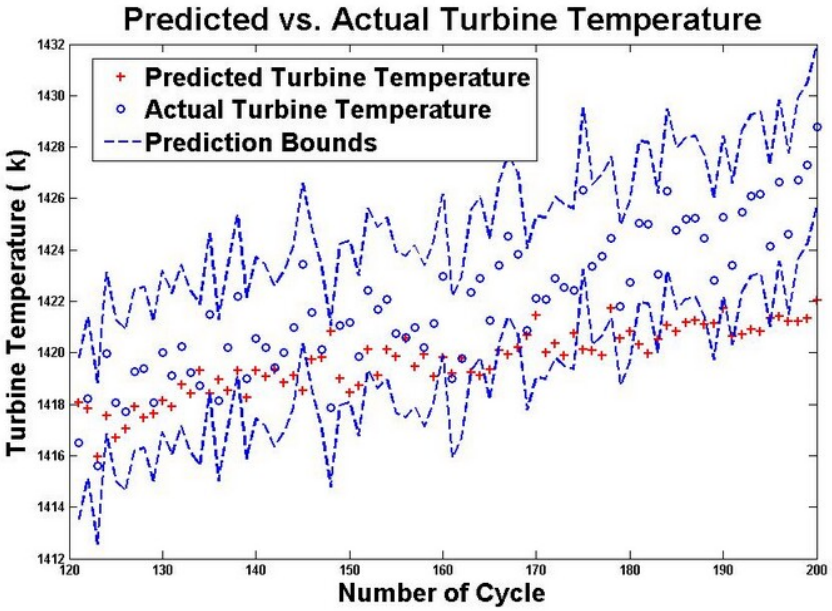

Figure 4.37: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $EI = 3\%$.

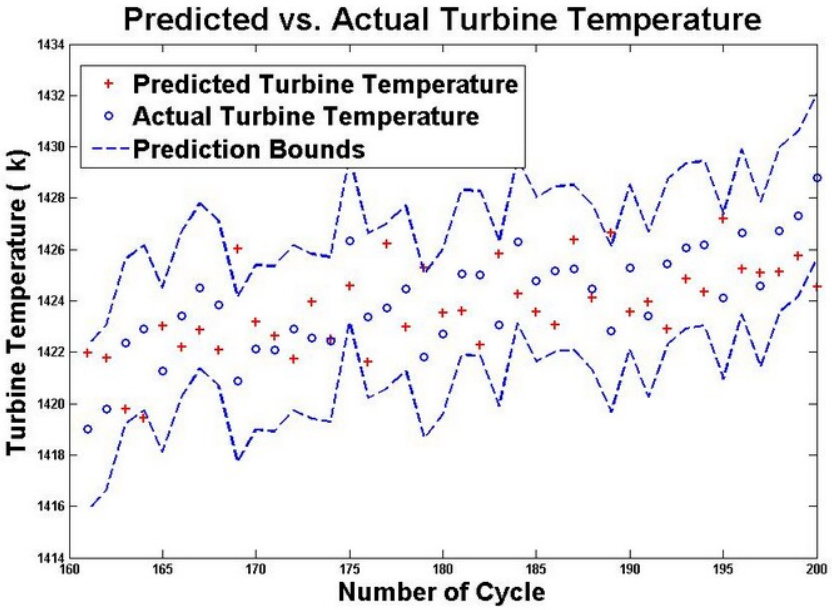

Figure 4.38: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $EI = 3\%$.

Table 4.35: A 2 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.0364	2.1201	3.6881
3	**0.2273**	**2.2120**	**2.1960**
4	1.1597	1.9591	2.2554
5	1.2454	2.2962	2.5868
6	2.2862	1.7762	2.8814
7	2.2359	1.8395	2.8807
8	2.2848	1.9898	3.0134

standard deviation, and RMSE are $2.5925K$, $2.8451K$, and $3.8404K$, respectively. As depicted in Figure 4.39, 65.83% of the predicted points are within the prediction bounds.

Table 4.36: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	0.9553	4.6483	4.7264
3	2.6786	3.4534	4.3591
4	**2.5925**	**2.8451**	**3.8404**
5	1.3972	3.7467	3.9841
6	3.4845	2.7600	4.4380
7	2.7472	4.0198	4.8551
8	3.7081	4.0276	5.4623

The RMSE decreases to $3.5999K$ for the network 3-5-1 when it is trained by using 120 data points as presented in Table 4.37. Comparing Tables 4.36 and 4.37, the RMSE decreases by 6.26%. The actual and predicted points are shown in Figure 4.40 along with the prediction bounds where 73.75% of the predicted points are within these bounds. The absolute differences between the actual and predicted values are also depicted in Figure 4.41.

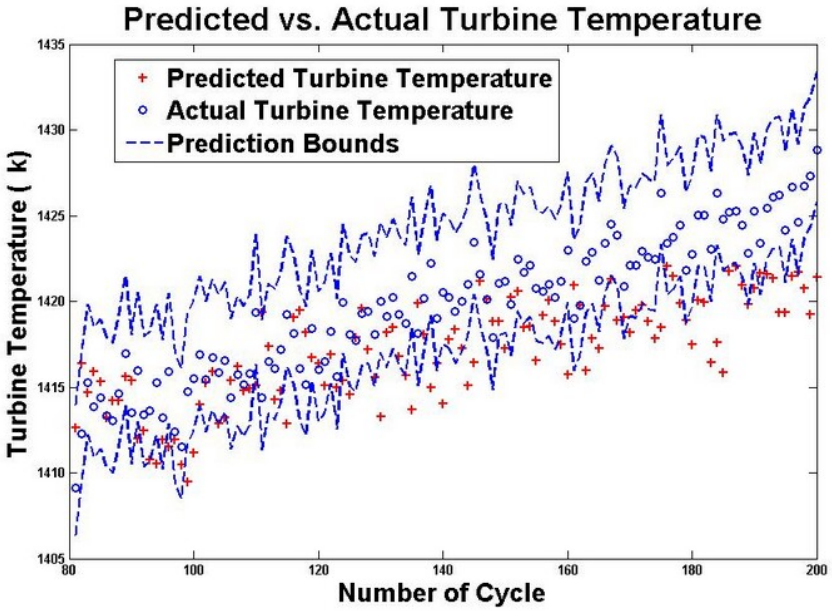

Figure 4.39: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $EI = 3\%$.

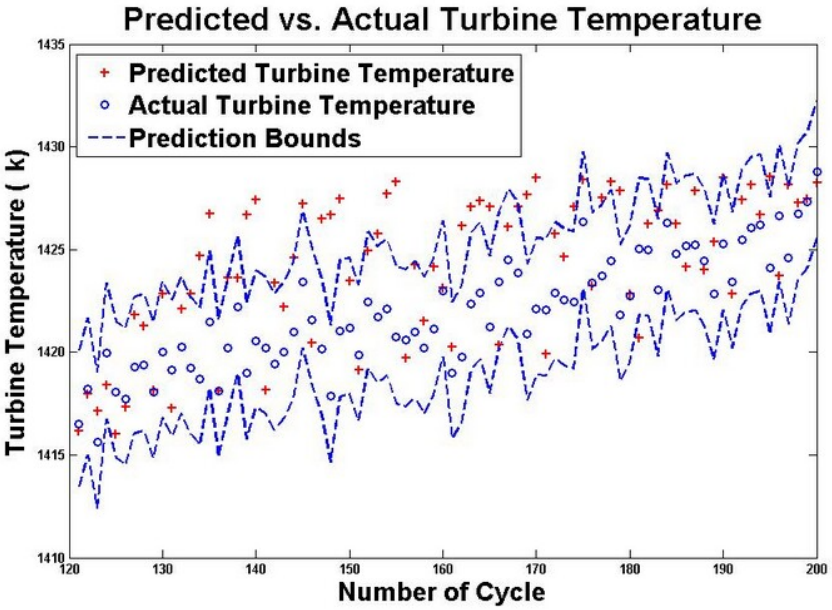

Figure 4.40: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $EI = 3\%$.

Table 4.37: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.9538	2.2588	4.5465
3	3.8675	2.2298	4.4573
4	3.3648	2.3123	4.0745
5	**2.2460**	**2.4311**	**3.5999**
6	3.0694	2.1975	3.7669
7	4.1733	2.4829	4.8481
8	4.2697	2.3735	4.8778

Next 80% of the entire available data are used to train the networks. The networks are then used to predict 5 flights ahead turbine temperature of 40 unseen data. The mean error, standard deviation and RMSE for these networks are summarized in Table 4.38. The network with 5 hidden neurons has the lowest RMSE. The predicted data for this network is shown pointwise in Figure 4.42 where 85% of the points are within the prediction intervals.

Table 4.38: A 5 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using Elman neural network

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.8287	2.2645	3.6057
3	2.1273	2.1881	3.0321
4	1.8348	2.3703	2.9739
5	**1.4544**	**1.8783**	**2.3569**
6	2.0098	1.9954	2.8145
7	3.0436	2.1152	3.6913
8	3.5869	2.1147	4.1504

For the next scenario, the Elman networks are trained and their performance are evaluated to investigate the reliability of this network in 8 flights ahead turbine temperature prediction. In the first case, 80 data points are used in the training

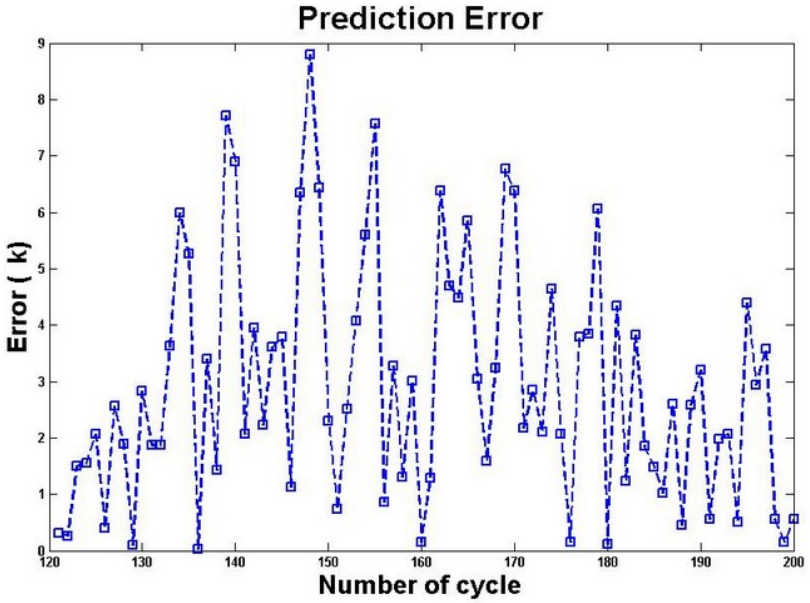

Figure 4.41: The prediction errors for the 5 step ahead turbine temperature when $EI = 3\%$ using the Elman 3-5-1 trained with 60% of the available data.

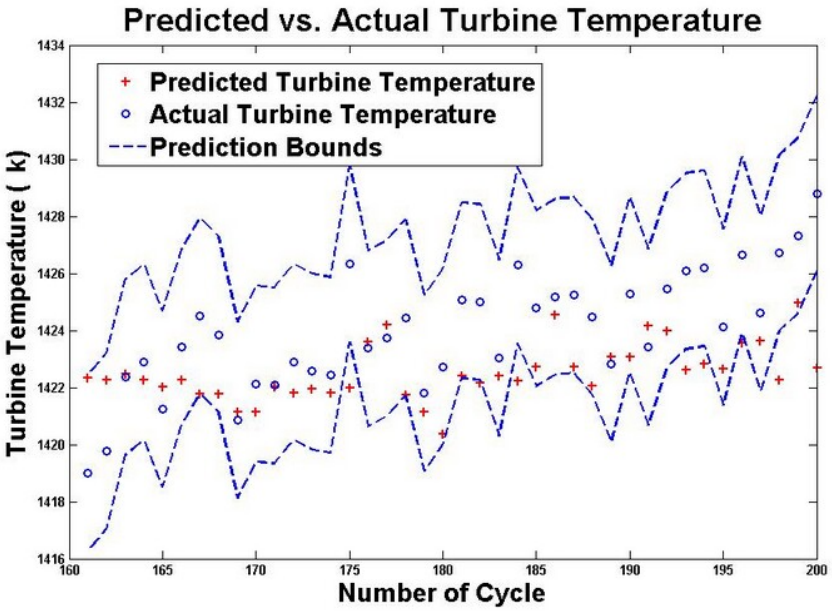

Figure 4.42: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $EI = 3\%$.

phase and 120 data are given to the networks with different architectures as inputs to predict the turbine temperature. The prediction errors of these networks are compared together in Table 4.39 where the network structure 3-4-1 has the best performance. In the second case, 120 data points are used to train the networks and the remaining 80 points to test the networks. The results of the prediction errors are shown in Table 4.40, and finally Elman networks with different number of hidden neurons are trained by using 160 available data. The networks then predict 40 unseen data and their errors are tabulated in Table 4.41.

Table 4.39: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.2111	4.0516	5.8320
3	4.1506	3.8326	5.6386
4	**3.3153**	**3.5972**	**4.8809**
5	3.3907	3.7828	5.0683
6	1.9441	4.8312	5.1890
7	3.6324	3.9445	5.3501
8	5.3755	3.5313	6.4235

Table 4.40: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.0110	3.2372	5.9547
3	4.2063	3.2401	5.2971
4	4.4640	2.3812	5.0524
5	**3.2773**	**3.1915**	**4.5606**
6	4.0957	2.2221	4.6530
7	4.6202	2.4800	5.2364
8	5.1179	2.9695	5.9077

The actual and predicted values for the optimal networks described previously

Table 4.41: An 8 flights ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.4859	2.3686	4.1978
3	2.7038	2.2592	3.5053
4	**2.5370**	**2.2113**	**3.3472**
5	2.9200	2.0888	3.5750
6	3.3334	2.3630	4.0688
7	3.5271	2.1275	4.1053
8	3.2557	2.5924	4.1415

are shown in Figures 4.43-4.45. Only 46.83% of the predicted values with the Elman network 3-4-1 trained by using 40% of the entire data sets are within the prediction bounds. This value increases to 76.25% when the Elman network 3-5-1 is trained with 60% of the available data points, and 80% when the network with 4 hidden neurons is trained by using 80% of the entire data points.

4.1.2.3 Summary of the Results

The optimal Elman neural networks are found in Section 4.1.2.1 to predict the turbine output temperature in presence of 1% erosion, and in Section 4.1.2.2 in presence of 3% erosion. The optimal networks are summarized for multi-flights ahead prediction in Tables 4.42 and 4.43, where N_{train} is the number of training data points and N_{test} is the number of data which were used to evaluate the trained network, and finally N_F is the number of flights ahead which the networks are used to predict the turbine temperature.

Based on Table 4.42, the RMSE decreases by the amount of 31.62% when the number of training data points increases by 50% for the Elman network to predict 2 flights ahead turbine temperature which emphasizes the importance of the training data sets in network performance. Moreover, the RMSE increases when the network

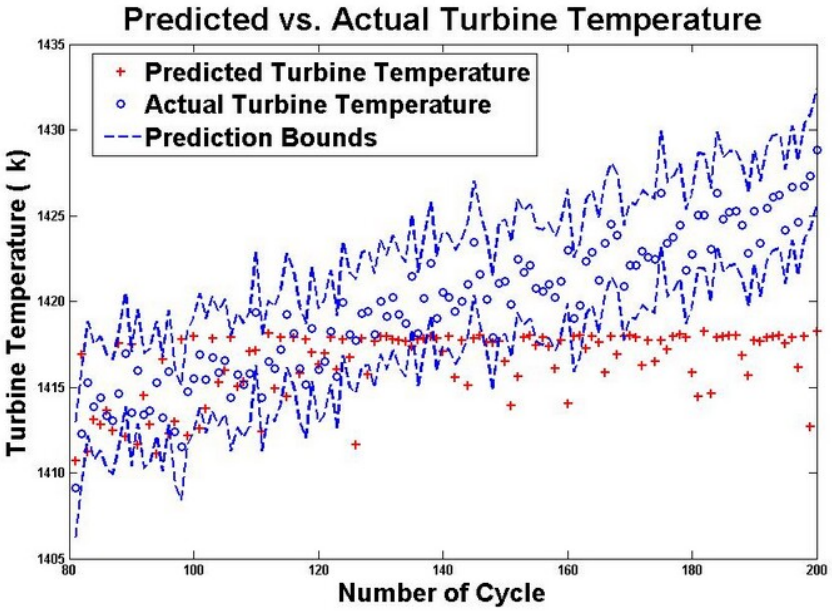

Figure 4.43: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $EI = 3\%$.

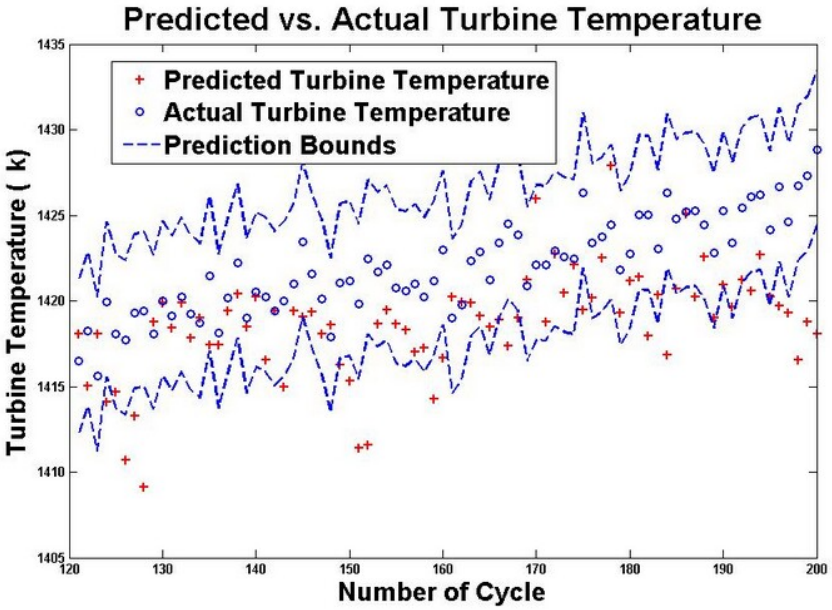

Figure 4.44: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $EI = 3\%$.

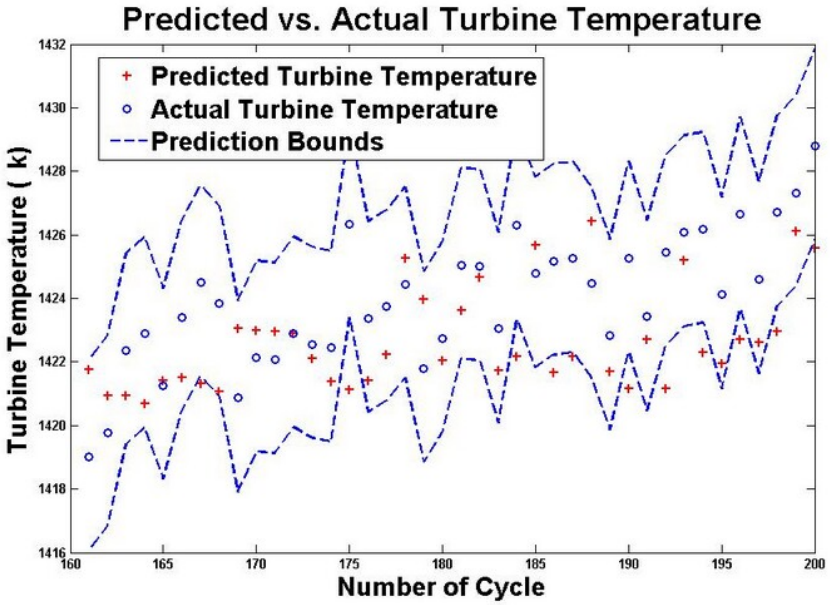

Figure 4.45: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $EI = 3\%$.

is used to predict more flights ahead with the same number of training data points as shown in Table 4.43 where the RMSE increases from $2.1960K$ to $3.3472K$ when the Elman network predicts 8 flights ahead turbine temperature instead of 2 flights ahead.

4.1.3 Concurrent Degradations

In this section, it is assumed that both fouling in the compressor and erosion in the turbine occur at the same time in the gas turbine engine. As described in Section 2.6.5 and equations (2.6.1)-(2.6.4), these degradations are modelled in our Simulink model. These degradations affect the performance of the engine. Based on Table 2.8, turbine temperature increases which causes an increase in the fuel flow rate. Therefore, it is important to predict the future health of the engine to reduce maintenance costs. Multi-flights ahead turbine temperature is predicted for different scenarios to

Table 4.42: Summary of the prediction errors for each scenario in presence of $EI = 1\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-4-1	2.5466
120	80	2	3-3-1	1.8621
160	40	2	3-3-1	1.4721
80	120	5	3-3-1	2.7976
120	80	5	3-4-1	2.0323
160	40	5	3-3-1	1.8563
80	120	8	3-5-1	3.2034
120	80	8	3-3-1	2.3273
160	40	8	3-3-1	2.2051
80	120	12	3-4-1	3.6699
120	80	12	3-4-1	2.5350
160	40	12	3-5-1	2.4037

investigate the reliability of the Elman neural network in long term prediction.

4.1.3.1 FI=1% and EI = 1%

The optimal Elman networks are found to predict multi-flights ahead turbine output temperature in presence of 1% compressor fouling and 1% turbine erosion. The efficiency of the compressor and turbine decreases by 1%. The compressor mass flow rate decreases by 0.5% due to adherence of particles to the blades, while the turbine mass flow rate increases by 0.5% due to the removal of materials from flow path in the turbine section of an engine. The entire data available equals to 200 points, and these data are used to train and evaluate the network for 2 flights ahead turbine temperature prediction. A total of 80 data sets are used to train different Elman networks. The trained networks are then tested with 120 data. The prediction errors are summarized in Table 4.44.

Based on Table 4.44, the network with 5 hidden neurons has the best performance to predict 2 flights ahead turbine temperature. The mean error, standard deviation

Table 4.43: Summary of the prediction errors for each scenario in presence of $EI = 3\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-4-1	3.2113
120	80	2	3-5-1	2.9329
160	40	2	3-3-1	2.1960
80	120	5	3-4-1	3.8404
120	80	5	3-5-1	3.5999
160	40	5	3-5-1	2.3569
80	120	8	3-4-1	4.8809
120	80	8	3-5-1	4.5606
160	40	8	3-4-1	3.3472

Table 4.44: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	7.7967	4.0355	8.7715
3	6.5263	5.3286	8.4113
4	6.0506	3.4838	6.9746
5	**4.1148**	**4.0672**	**5.7737**
6	5.6754	3.7180	6.7763
7	6.4345	4.7575	7.4434
8	7.4894	3.9419	8.4558

and RMSE are $4.1148K$, $4.0672K$,and $5.7737K$ respectively. The actual and predicted data are depicted pointwise in Figure 4.46 where 61.66% of the predicted points are within the upper and the lower prediction bounds.

By increasing the number of training data sets to 120, the Elman network with 5 hidden neurons has the lowest RMSE. The comparison of prediction errors among various Elman network structures are presented in Table 4.45. The predicted and actual values are shown in Figure 4.47. The prediction intervals are also depicted to overcome the uncertainty problem where 58.75% of the predicted points are between these bounds.

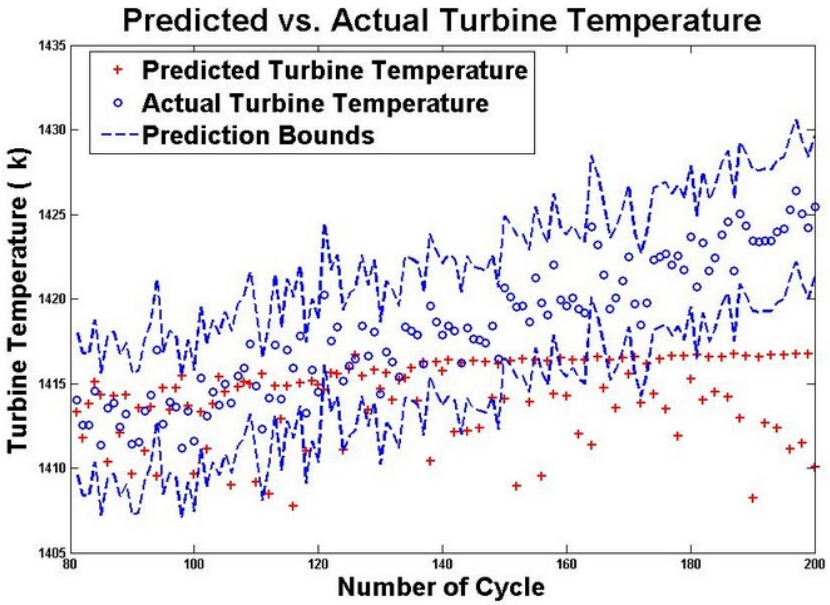

Figure 4.46: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

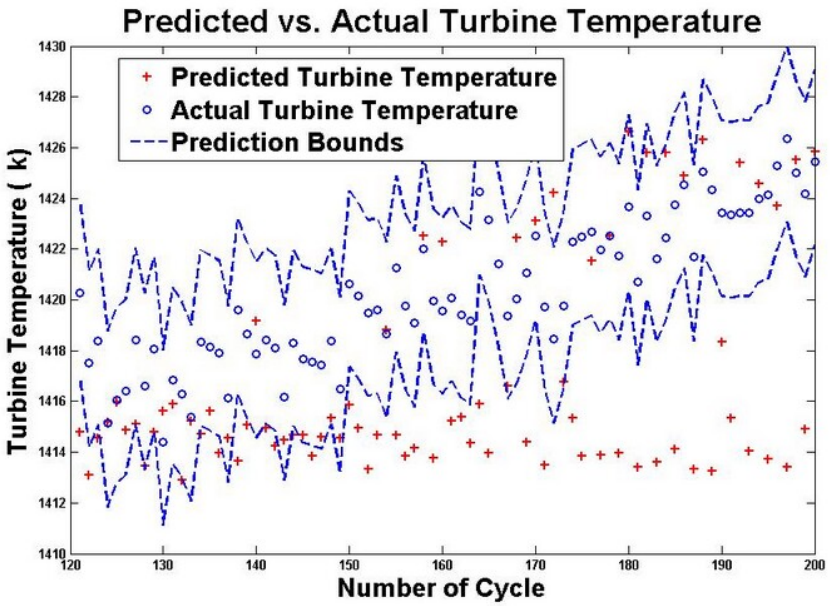

Figure 4.47: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 4.45: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	6.1118	4.8770	6.7474
3	5.8293	4.9921	6.5438
4	5.0066	3.1942	5.4608
5	**3.4052**	**2.9470**	**5.1942**
6	4.6696	3.7691	5.4201
7	5.7958	4.7724	6.4173
8	6.2362	4.6377	6.7647

Next 160 data points are used in the training phase. The number of hidden neurons are changed and the trained networks are tested with 40 unseen data. The networks output which is the predicted turbine temperature for 2 flights ahead are compared to the actual ones, and the mean, standard deviation and RMSE of these networks are summarized in Table 4.46. The actual and predicted points for the Elman network 3-3-1 are shown in Figure 4.48 where only 7.5% of the predicted points are outside the prediction intervals.

Table 4.46: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	1.4533	2.8847	3.1977
3	**2.0799**	**1.9849**	**2.8579**
4	2.3353	2.0632	3.0990
5	2.6940	1.8796	3.2714
6	2.7442	2.0118	3.3878
7	2.8261	2.6069	3.8226
8	3.7595	2.0301	4.2606

A 5 flights ahead turbine temperature is predicted using the Elman neural network under three different cases. The entire available data set is 200. First, 40% of the

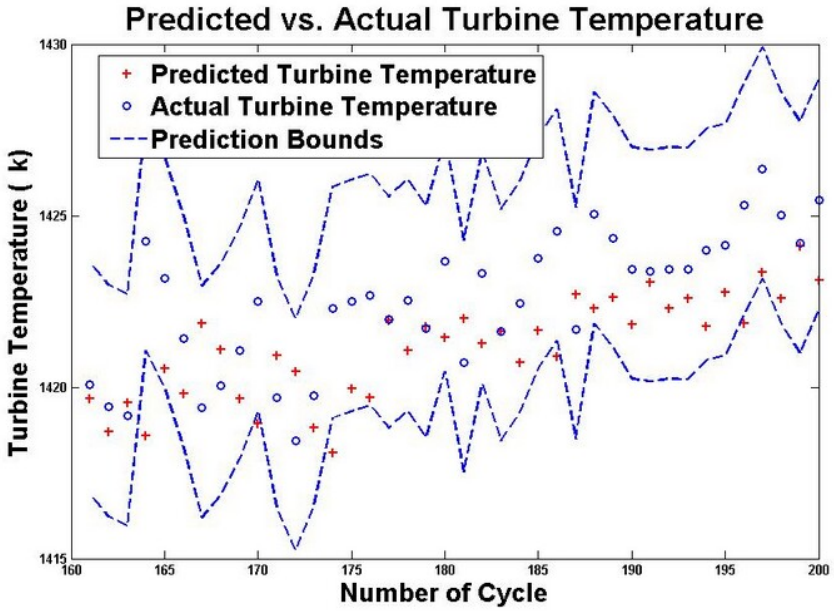

Figure 4.48: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

available data is used in the training phase. Different Elman network structures are trained. The weights and biases are fixed and these networks are evaluated by using 80 data points. The prediction error for these networks are presented in Table 4.47 where the network with 4 hidden neurons has the lowest RMSE. The predicted data for this network are shown in Figure 4.49 along with the prediction bounds where only 54.17% of the predicted data are within the bounds.

In the second case 60% of the entire data points are used in the training phase and 40% for the network evaluation. The statistical error measures for the Elman networks with different architectures are presented in Table 4.48. The network with 5 hidden neurons has the best performance in prediction. The mean, standard deviation, and RMSE during testing phase are $4.6767K$, $3.3411K$, and $5.6227K$, respectively. The predicted values for this network are shown in Figure 4.50 where 63.75% of the predicted data are within the prediction intervals.

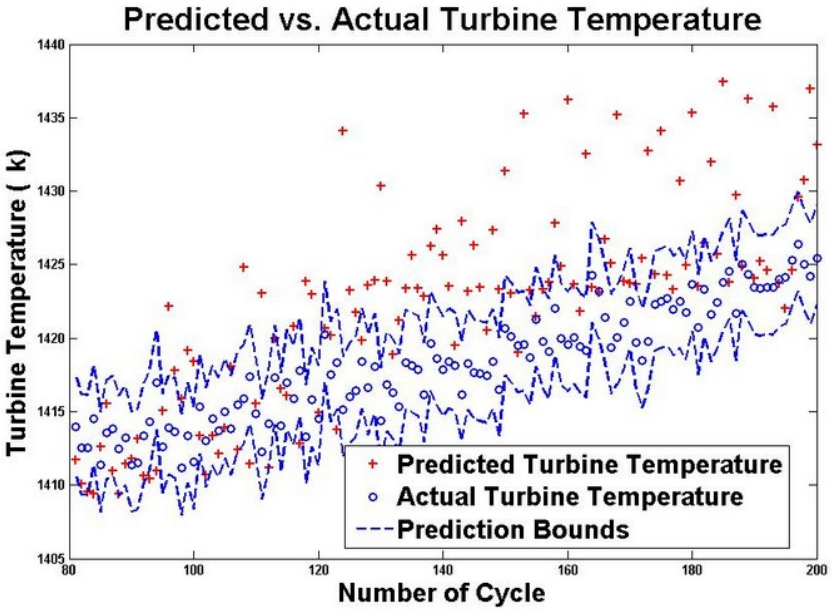

Figure 4.49: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

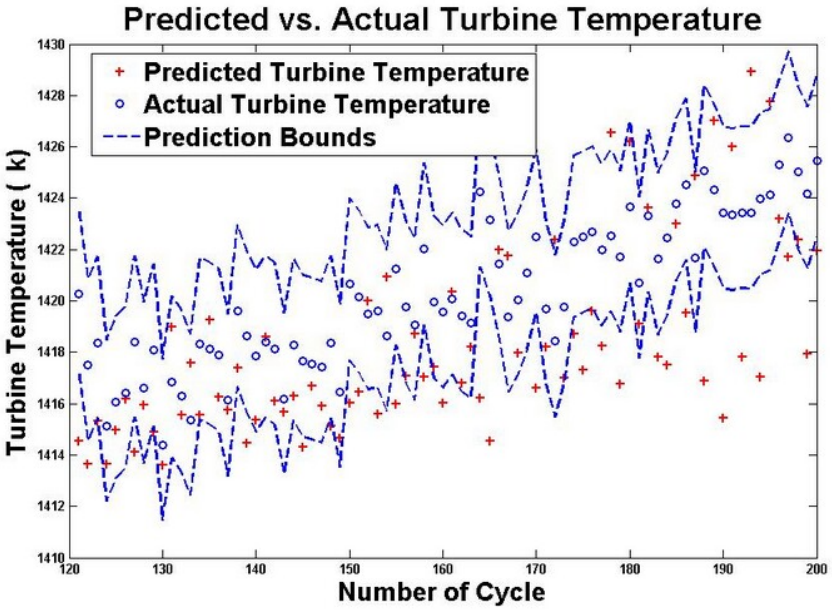

Figure 4.50: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 4.47: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	6.2283	4.9385	7.9358
3	5.9884	4.9427	7.1607
4	**4.0647**	**5.2907**	**6.6543**
5	5.9259	5.4716	6.8606
6	6.8877	4.9289	7.9213
7	7.4555	5.1972	8.5472
8	7.7421	5.7884	8.6124

Table 4.48: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.8268	5.0556	6.5705
3	5.6780	4.7286	6.2922
4	5.0947	4.0183	5.9121
5	**4.6767**	**3.3411**	**5.6227**
6	5.4340	4.7873	6.0992
7	5.9604	4.4649	6.4441
8	6.3741	5.0993	7.0792

Finally, in the third case, Elman networks are trained with 80% of the entire data points and tested with the remaining 20% of the data. The results of the prediction error are tabulated in Table 4.49. The actual and predicted values for the Elman network 3-5-1 are depicted in Figure 4.51 where 92.5% of the predicted points are between the upper and the lower prediction bounds.

Different Elman neural network structures are trained and tested to investigate the applicability of the Elman network to predict 8 flights ahead turbine output temperature. A total number of 80 data sets are used in the training phase and these networks are then tested to predict 120 data. The summary of the prediction errors

Table 4.49: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.0996	3.1008	4.3569
3	3.1581	2.3252	3.9045
4	2.9105	2.2540	3.6640
5	**2.5491**	**1.1230**	**3.3003**
6	2.5507	2.4512	3.5163
7	3.9935	2.4088	4.6482
8	4.2441	3.3961	4.8590

are shown in Table 4.50. When the number of training data points is increased to 120, the results of the prediction error are presented in Table 4.51, and finally the prediction error for the networks trained by using 160 data points and evaluated with 40 data are presented in Table 4.52.

Table 4.50: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	7.8829	5.8387	8.7609
3	7.3827	4.5589	8.1893
4	**6.5183**	**5.7355**	**7.5050**
5	7.0811	5.7407	8.0012
6	7.5220	5.5246	8.7682
7	8.3073	5.8244	9.1387
8	8.4650	6.9634	9.3399

Figure 4.52 shows the predicted turbine temperatures when the Elman network 3-4-1 trained by using 80 data points where 56.67% of the predicted points are within the prediction bounds. Predicted values of the network 3-5-1 which is trained by using 120 data points are shown in Figure 4.53 where 78.75% of the predicted values are within the upper and the lower bounds, and finally 90% of the predicted values

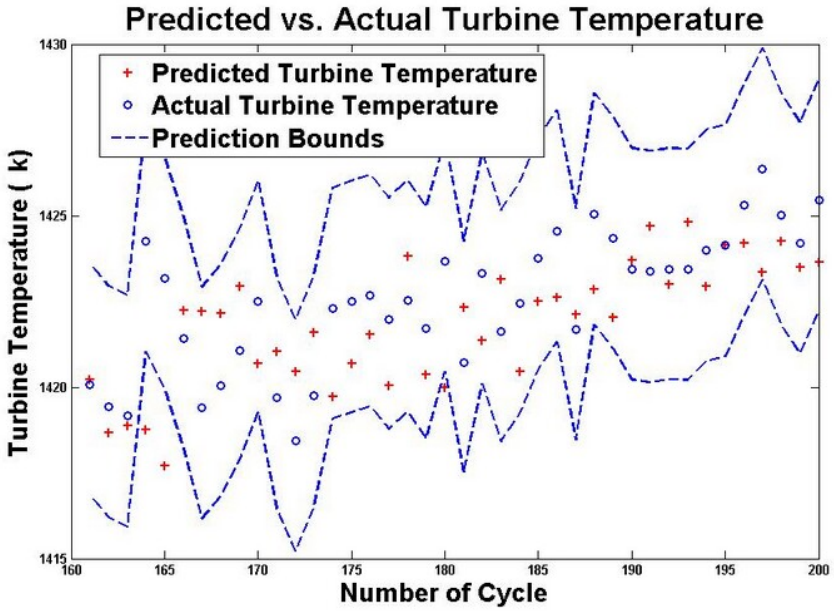

Figure 4.51: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

of the network 3-3-1 trained with 160 data points are within the prediction intervals as shown in Figure 4.54.

4.1.3.2 FI=3% and EI = 2%

The reliability of the Elman neural network in multi-step ahead turbine temperature prediction is investigated in presence of 3% compressor fouling and 2% turbine erosion. The compressor efficiency degrades by 3% and the turbine efficiency degrades by 2%. Due to this deterioration, the compressor mass flow rate decreases by the amount of 1.5% and the turbine mass flow rate increases by 1%. This degradation is modelled in our Simulink model as described in Section 2.6.5. It is assumed that the engine goes through this degradation in over 200 flights. Turbine temperature is predicted for 2 flights ahead. A total of 80 data sets are used to train the Elman networks and 120 ones are used in the testing phase. A comparison of the prediction error for different

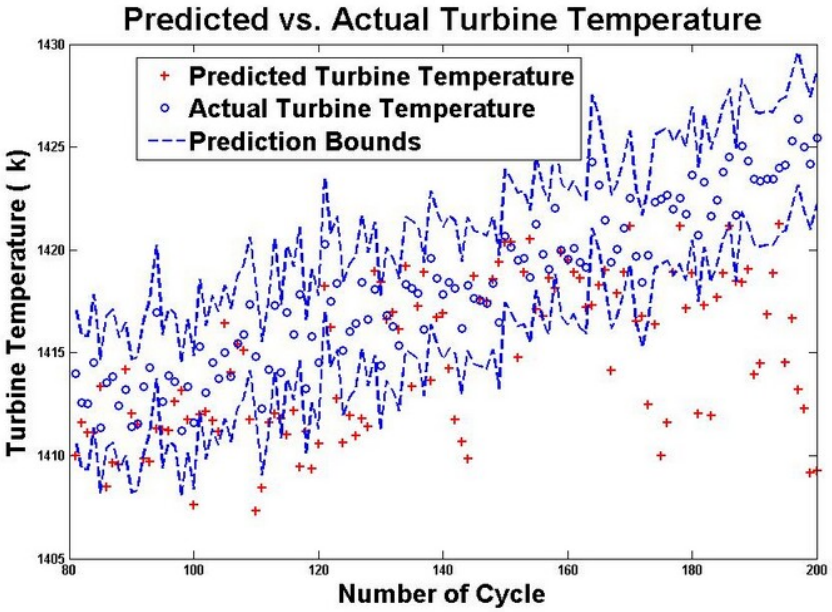

Figure 4.52: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 1\%$ and $EI = 1\%$.

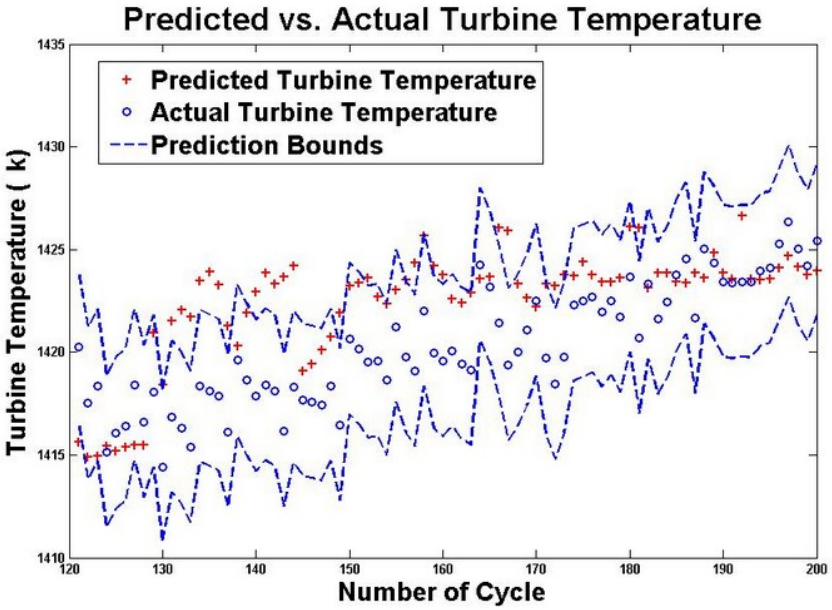

Figure 4.53: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$.

Table 4.51: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	6.5760	4.1223	7.2712
3	6.2754	5.6145	6.7919
4	6.0641	4.7921	6.6688
5	**5.6791**	**4.7611**	**6.2444**
6	5.8427	3.8494	6.4927
7	5.9752	4.0472	7.2027
8	6.7890	5.7668	7.7525

Table 4.52: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.6052	2.1179	5.0578
3	**3.6561**	**3.0559**	**4.1819**
4	3.6147	3.3550	4.2981
5	3.8170	3.6848	4.6473
6	4.2644	3.1708	4.7728
7	4.1435	3.7586	4.9586
8	5.0144	3.6416	5.6523

network architectures are shown in Table 4.53.

Based on Table 4.53, the Elman network with 6 hidden neurons has the lowest RMSE. The actual and predicted data of this network are shown in Figure 4.55 where only 39.17% of the predicted data are within the upper and the lower prediction intervals.

Next the number of training data sets is increased to 120 points and the networks with different number of hidden neurons are trained. These networks are then evaluated with 80 unseen data. The prediction errors are summarized in Table 4.54 for these networks. The predicted values for the network 3-5-1 are depicted in Figure

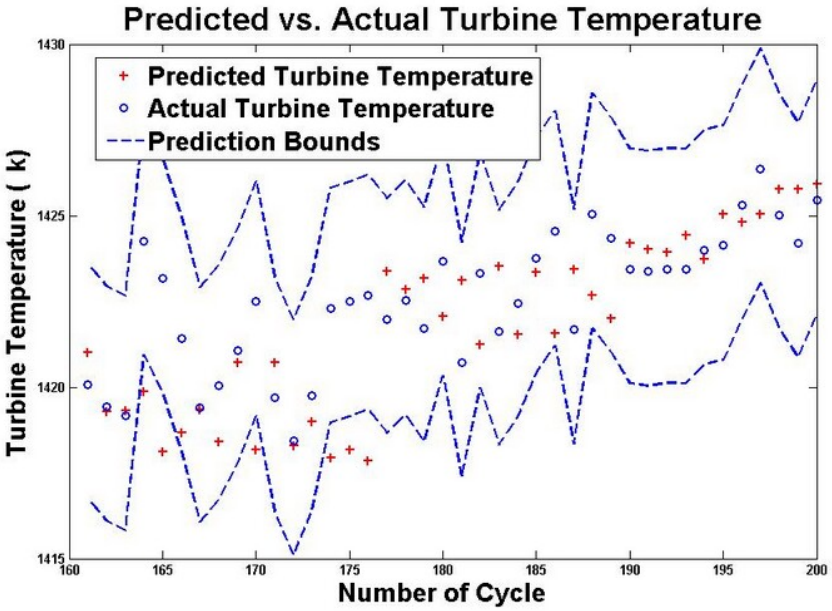

Figure 4.54: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $FI = 1\%$ and $EI = 1\%$.

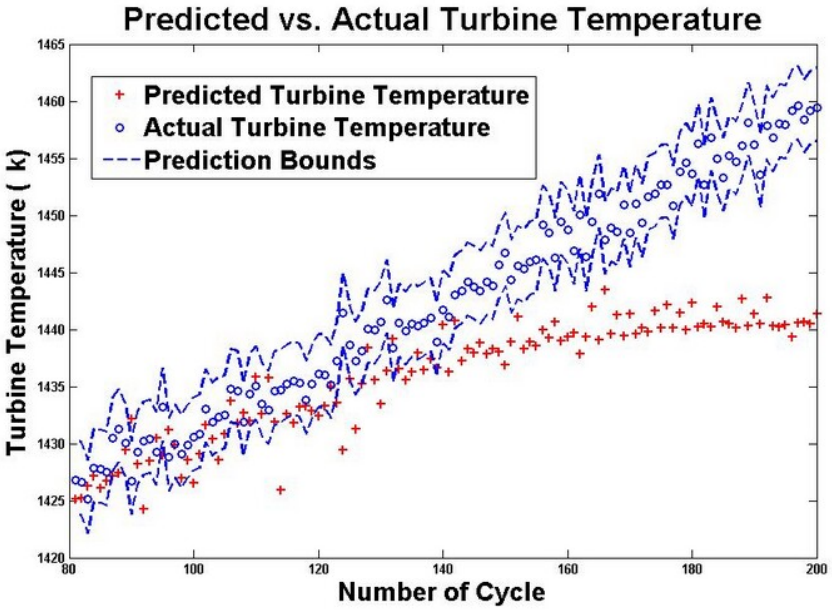

Figure 4.55: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-6-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 4.53: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	8.8630	8.2584	12.0907
3	8.4562	8.8843	11.5701
4	3.6856	8.8616	9.5633
5	7.9072	5.5022	9.0898
6	**6.6141**	**4.6846**	**8.7059**
7	5.8079	6.6092	8.7777
8	8.0963	5.6931	9.8839

4.56 where 47.5% of the predicted values are within the prediction intervals.

Table 4.54: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.8748	6.9415	9.0606
3	7.8104	6.2795	8.8931
4	6.5060	5.0708	7.6611
5	**6.3884**	**5.2956**	**7.1789**
6	6.7413	4.3843	8.0266
7	8.0554	5.8064	9.3649
8	10.3323	10.2666	12.0638

When the networks are trained by using 160 data points, the network with 4 hidden neurons has the lowest RMSE equal to $3.2062K$ as shown in Table 4.55. The actual and predicted points for this network are depicted pointwise in Figure 4.57. To overcome the problem of uncertainty in measurements the upper and the lower prediction bounds are also shown where 70% of the predicted values are within these bounds.

A 5 flights ahead turbine temperature is predicted using the Elman network. Also 40% of the entire data points which is equal to 80 data are used to train the networks.

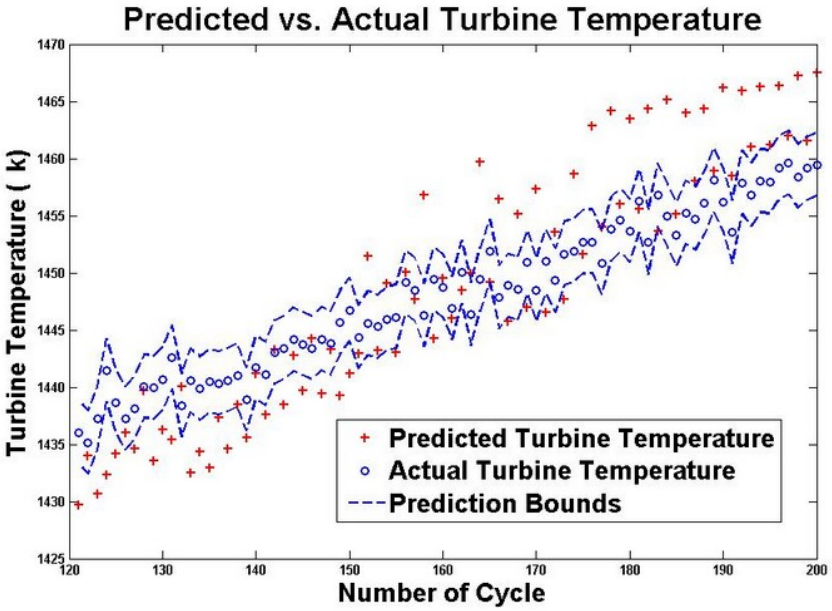

Figure 4.56: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$.

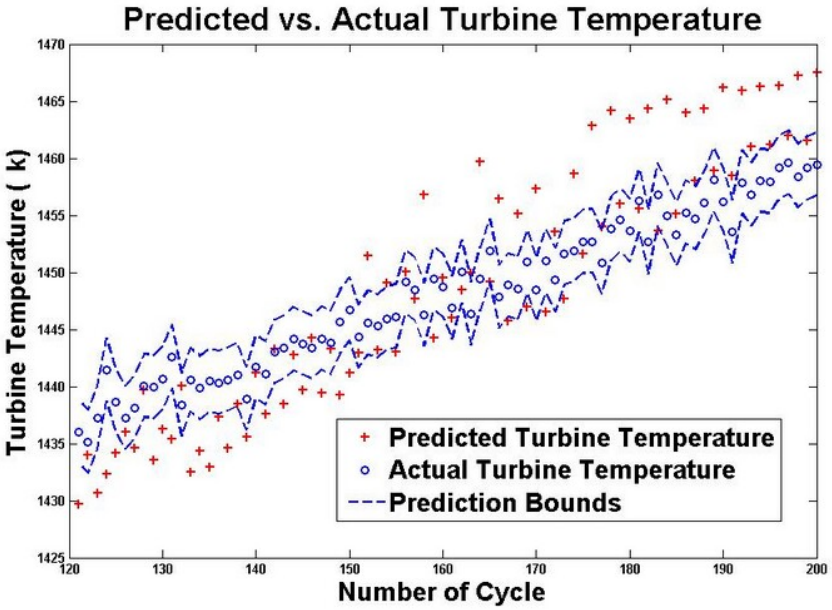

Figure 4.57: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 4.55: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.0396	3.7581	4.8719
3	3.1827	2.7049	4.1549
4	**2.5514**	**1.9663**	**3.2062**
5	2.9799	2.1257	3.6450
6	3.0128	3.1168	4.3068
7	3.8794	3.8912	4.8166
8	4.0313	3.8362	4.9086

The number of hidden neurons are changed from 2 to 8. The trained networks are then evaluated by using 120 data. The statistical error measures during testing phase are summarized in Table 4.56. The network with 5 hidden neurons has the lowest error. The actual and predicted data are shown pointwise in Figure 4.58 where 44.1% of the predicted points are within the prediction intervals.

Table 4.56: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	12.6527	8.7047	14.7973
3	0.2152	14.2474	14.1895
4	11.5619	8.6939	13.8701
5	**7.9979**	**8.1988**	**11.4291**
6	2.0911	12.6356	12.7554
7	10.5910	9.8166	13.1438
8	12.4005	9.6799	14.5692

Next 60% of the available data points are used in the training phase and the networks are tested with the remaining 40% data. The networks with different structures are trained and their performance in 5 flights ahead turbine temperature prediction is compared together in Table 4.57. The network with 5 hidden neurons has the best

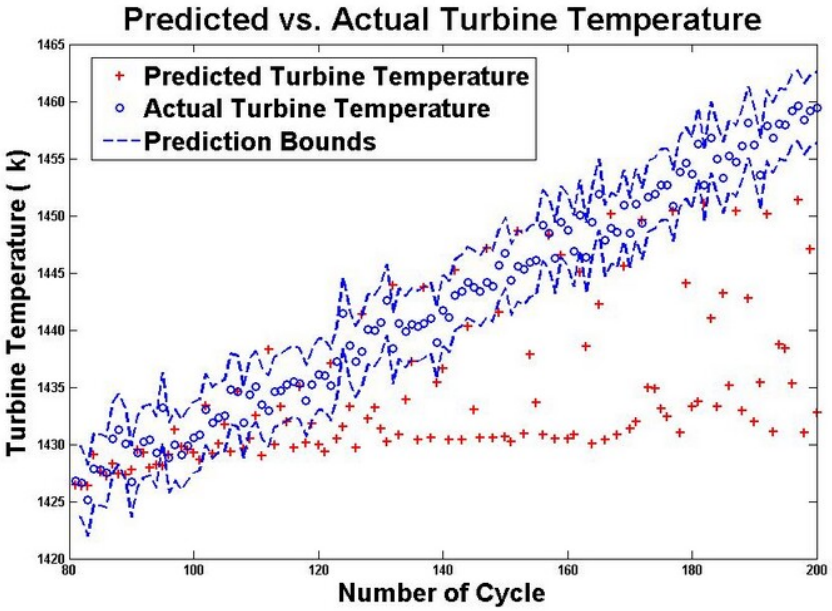

Figure 4.58: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 2\%$.

performance. The predicted values of this network are depicted in Figure 4.59 along with the actual data and prediction bounds where only 32.5% of the predicted points are between the upper and the lower prediction bounds.

Finally, the number of training data sets increases to 80% of the entire data. Seven different Elman network structures are trained with these data. The networks are then used to predict 5 flights ahead turbine temperature. The differences between the actual and the network's output are calculated and the errors are summarized in Table 4.58 where the network with 4 hidden neurons has the lowest RMSE. The outputs of this network are depicted in Figure 4.60 along with the actual temperatures and prediction bounds where 77.5% of the predicted turbine temperatures are within the prediction bounds.

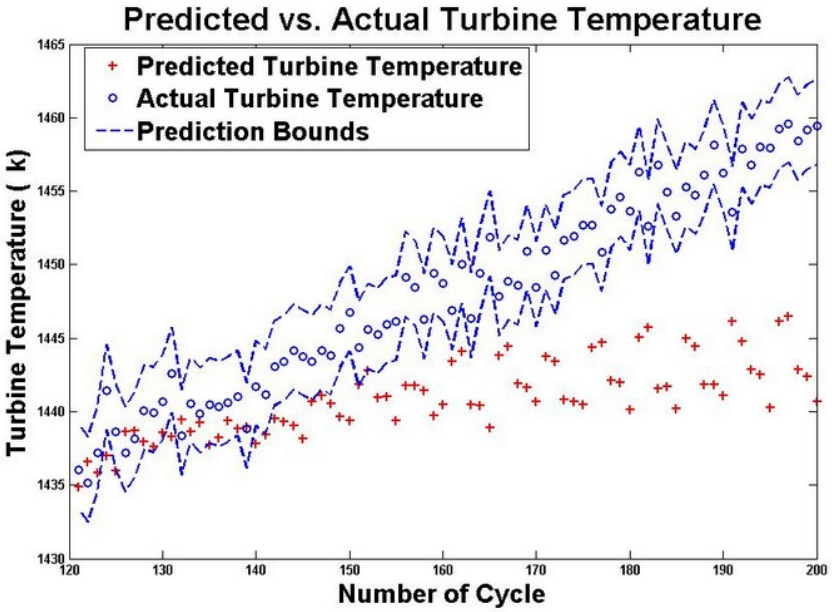

Figure 4.59: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$.

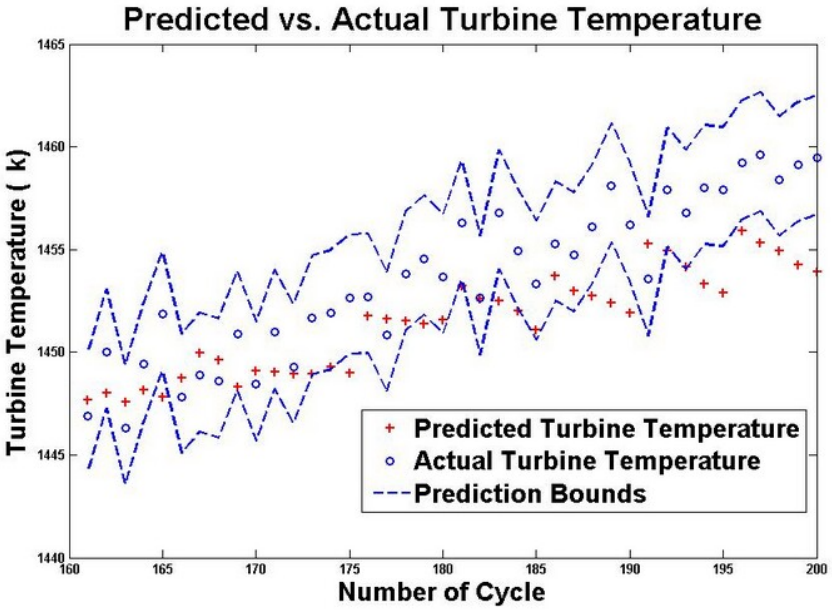

Figure 4.60: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$.

Table 4.57: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	9.6321	6.6395	11.1438
3	9.3996	5.6849	10.9666
4	7.7219	5.3091	9.3521
5	**7.1577**	**5.1450**	**8.7961**
6	7.5014	4.9824	8.9881
7	9.6420	6.0456	11.3605
8	10.4984	7.7584	11.9566

Table 4.58: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.0918	4.0369	4.5016
3	2.7761	2.7782	3.9028
4	**2.6687**	**2.6793**	**3.7578**
5	2.8205	2.7801	3.9358
6	3.2786	2.4964	4.1019
7	3.6132	3.2711	4.8464
8	3.9551	3.1391	5.0250

4.1.3.3 FI=2% and EI = 3%

The turbine goes through 2% fouling in 200 flights which implies that the compressor efficiency decreases by the amount of 2% and its mass flow rate decreases by the amount of 1%. In the same time the turbine eroded by the index of 3% which implies a reduction of 3% in the turbine efficiency and a 1.5% increase in its mass flow rate. Data is generated in our Simulink model described in Section 2.5 based on equations (2.6.1)-(2.6.4). As done previously in Section 4.1.3.2, the entire data set is divided into two parts; one part is used for training the networks and the other part is used to evaluate the performance of the networks in multi-step prediction.

Turbine temperature is predicted for 2 flights ahead using three different training and testing data sets. First, 80 data points are used in the training phase and 120 data are used in the evaluation phase. Networks with different structures are trained and their performance are compared together in Table 4.59.

Table 4.59: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	8.0711	5.8208	8.9229
3	5.9866	4.9807	7.1801
4	1.0283	6.7492	6.7993
5	**0.6764**	**4.7356**	**6.2641**
6	4.8687	4.8680	6.8706
7	6.2466	4.0767	7.4499
8	4.9268	6.0001	7.7443

Based on Table 4.59, the network with 5 hidden neurons has the lowest error in predicting turbine temperature. The predicted values of this network are shown in Figure 4.61 along with the actual values and prediction bounds where 73.33% of the predicted points are within the bounds.

Second, the number of training data sets is increased to 120 points. The number of hidden neurons are changed from 2 to 8 and these networks are tested by using the remaining 80 data. The summary of the prediction errors is presented in Table 4.60. The Elman network 3-5-1 has the best performance. The mean, standard deviation, and RMSE are $3.8652K$, $3.4199K$, and $5.3468K$, respectively. The actual and predicted turbine temperatures for the network 3-5-1 are shown in Figure 4.62 where 77.5% of the predicted points are within the prediction intervals.

Finally, 160 data sets are used in the training phase and 40 data are used in the testing phase. As can be seen in Table 4.61, the network structure 3-3-1 has the best performance in 2 flights ahead prediction. From Figure 4.63, only 22.5% of the

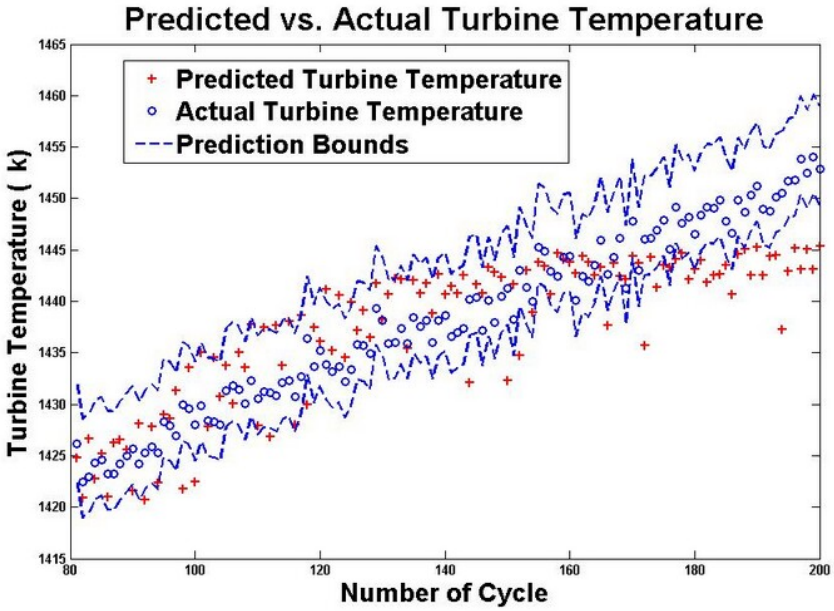

Figure 4.61: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

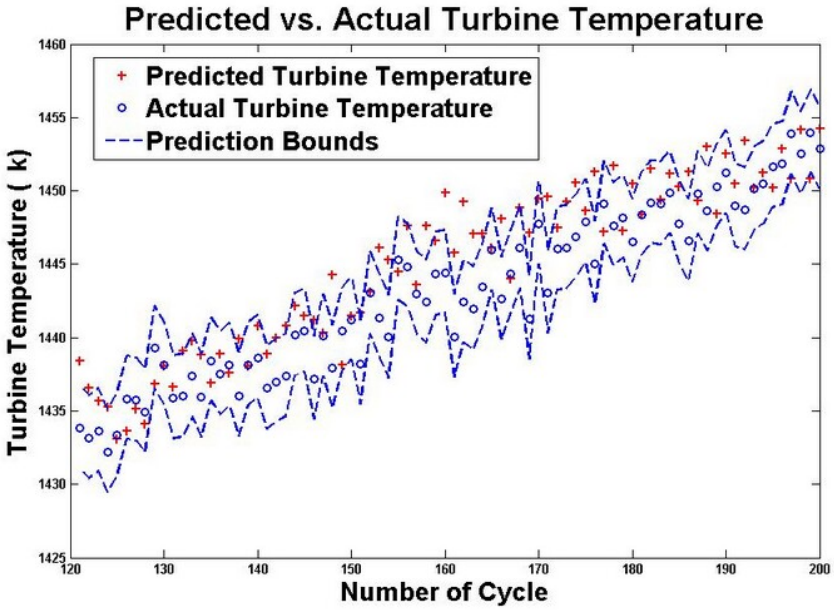

Figure 4.62: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 4.60: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.1915	4.6292	6.3212
3	4.8148	3.7585	6.0936
4	4.7437	3.6655	5.9809
5	**3.8652**	**3.4199**	**5.3468**
6	5.0226	3.4772	6.0965
7	5.7938	4.4818	6.7483
8	5.7613	4.0403	7.0223

predicted points for this network are outside the bounds.

Table 4.61: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.1824	2.2490	3.1136
3	**1.1941**	**2.8238**	**3.0333**
4	1.8942	2.5914	3.1836
5	2.4971	2.2160	3.3201
6	3.0801	2.2261	3.7840
7	2.8747	2.9658	4.1036
8	3.3372	2.7747	4.3178

A 5 flights ahead trubine temperature is now predicted through 3 different cases using the Elman neural network. In the first case, various network architectures are trained with 80 data sets. These networks are then evaluated by using 120 data. The prediction errors are tabulated in Table 4.62 where the network structure 3-4-1 has the best performance. In the second case the number of training points are increased to 120 and the performance of the networks are compared together by the remaining 80 data. The comparison of the prediction error among these networks are presented in Table 4.63, and finally 160 data points are used in the training phase, while the

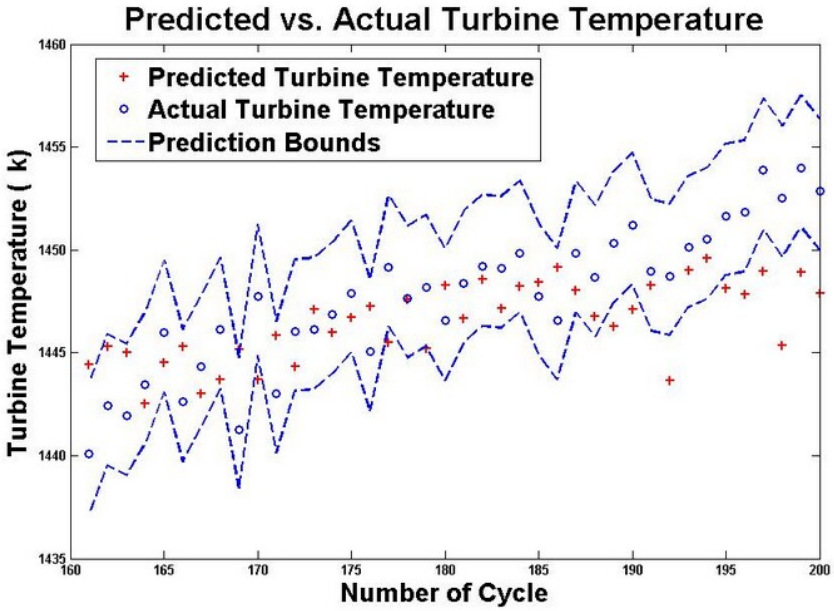

Figure 4.63: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

networks are tested by using 40 unseen data. The summary of the statistical errors of prediction can be seen in Table 4.64 where the network 3-5-1 has the lowest RMSE.

The predicted values for the optimal networks found in Tables 4.62-4.64 are depicted in Figures 4.64-4.66. For the Elman network 3-4-1 trained by using 80 data points 74.17% of the points are within the prediction bounds. This value increases to 76.25% when the network 3-3-1 is trained with 120 data points, and finally 95% if the network 3-5-1 is trained with 160 data sets as shown in Figure 4.66.

The Elman network now is used to predict 8 flights ahead turbine temperature as done previously. A total number of 80 data points are used to train different structures and these structures are evaluated by using 120 unseen data. The prediction errors are shown in Table 4.65. The actual and predicted values for the network 3-6-1 are depicted pointwise in Figure 4.67 along with the upper and the lower prediction bounds where 39.17% of the predicted values are between these bounds.

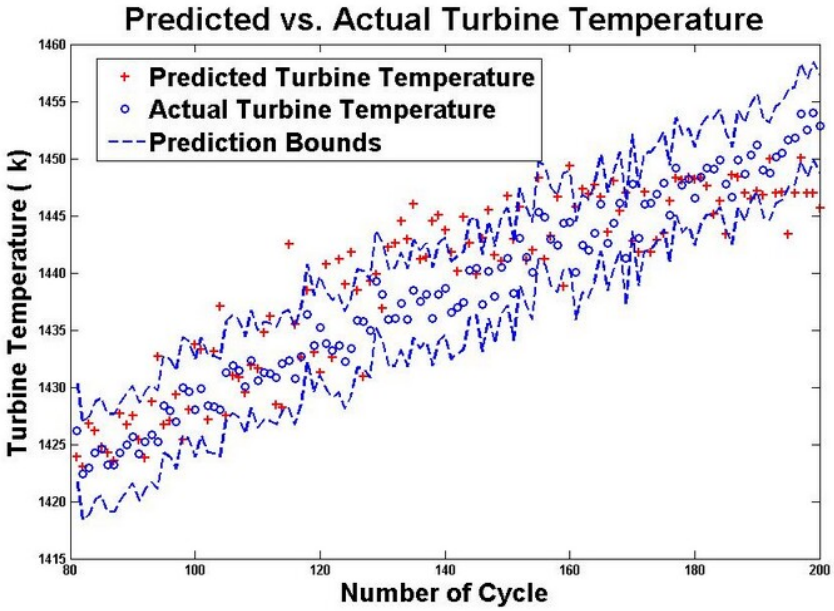

Figure 4.64: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

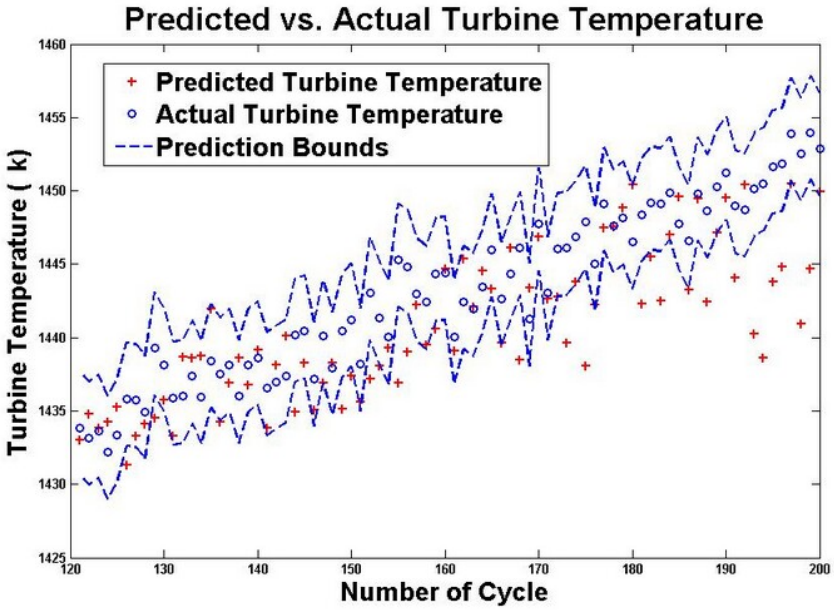

Figure 4.65: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-3-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

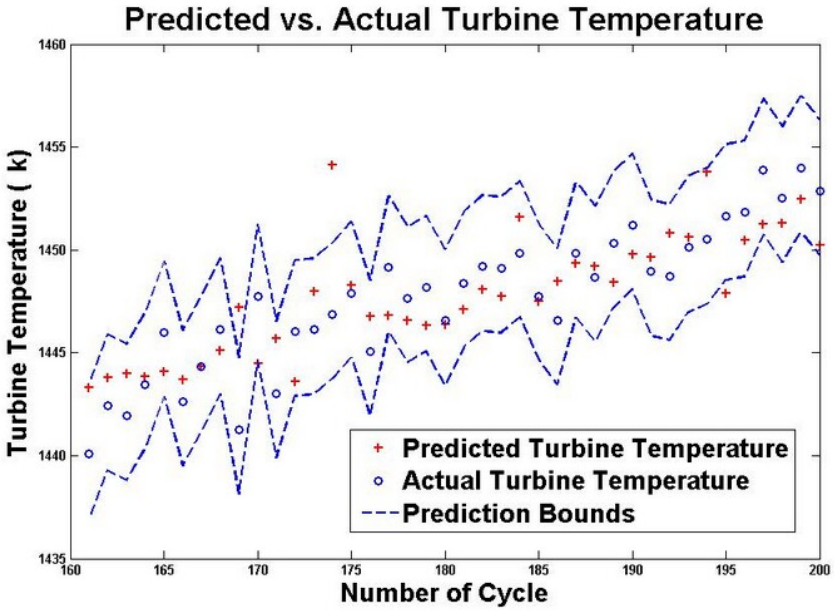

Figure 4.66: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

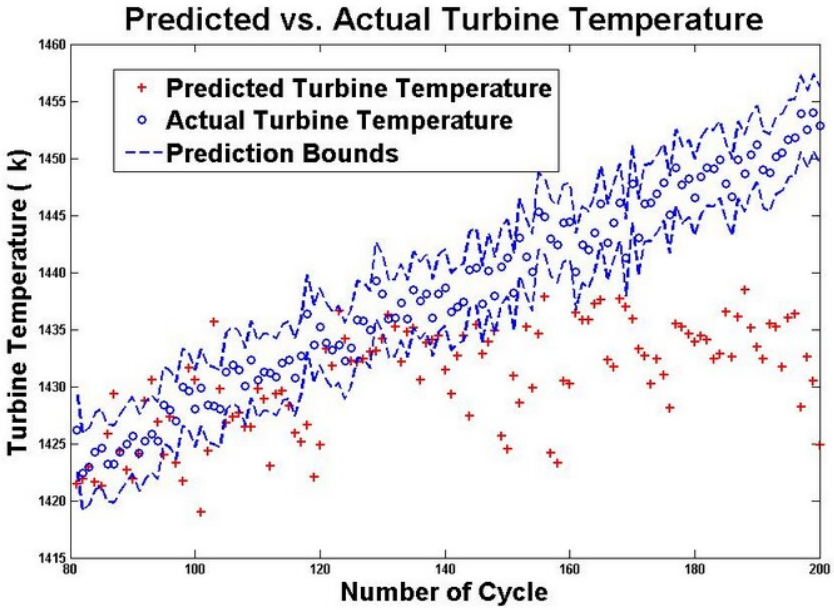

Figure 4.67: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-6-1 trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 4.62: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.6536	7.4967	9.3646
3	6.7013	5.7415	8.8089
4	**3.3736**	**6.5438**	**7.3380**
5	2.2206	7.2106	7.5160
6	4.9378	7.0162	8.5556
7	5.1583	7.5647	9.1300
8	6.9449	6.4089	9.4320

Table 4.63: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	5.4737	5.0543	7.4289
3	**3.7161**	**4.7873**	**6.0366**
4	3.4097	5.4115	6.3674
5	5.6589	5.3514	7.7655
6	7.0202	5.8909	8.5385
7	7.1277	5.1758	8.7896
8	6.9979	5.1518	8.6707

Next 7 different Elman network architectures are trained with 120 data sets. The weights and biases are fixed. The remaining 80 data are given as unseen inputs to the networks and the predictability of these networks for 8 flights ahead turbine temperature are compared together in Table 4.66. The actual and the predicted data for the network with 4 hidden neurons are depicted pointwise in Figure 4.68 where 61.25% of the predicted points are within the prediction intervals.

By increasing the number of training data sets to 160, the best Elman network performance is achieved when the network has 5 hidden neurons. The results of the prediction error for different number of hidden neurons are summarized in Table 4.67.

Table 4.64: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	2.4798	3.2331	4.0424
3	2.7526	2.5949	3.7606
4	2.1071	3.0084	3.6420
5	**2.1718**	**2.6081**	**3.3688**
6	2.8924	2.9070	4.0749
7	2.9802	2.8258	4.0826
8	2.4372	4.0326	4.6686

Table 4.65: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	14.9085	10.8718	17.8579
3	14.0705	10.7831	16.0640
4	9.9017	9.4105	13.6332
5	7.9760	8.1151	11.3544
6	**7.5660**	**6.9366**	**10.2450**
7	12.5919	10.8934	14.8440
8	15.1500	12.0836	17.6451

Figure 4.69 shows the predicted turbine temperatures of the network 3-5-1 where 80% of the predicted values are within the upper and the lower prediction intervals.

4.1.3.4 FI=3% and EI = 3%

In the most severe scenario the engine has 3% fouling in the compressor and at the same time 3% erosion in the turbine. The compressor and turbine efficiency both degrade by the amount of 3%. The compressor mass flow rate decreases by the amount of 1.5%, while the turbine mass flow rate increases by this amount. The degraded data are generated by using the engine Simulink model as mentioned in

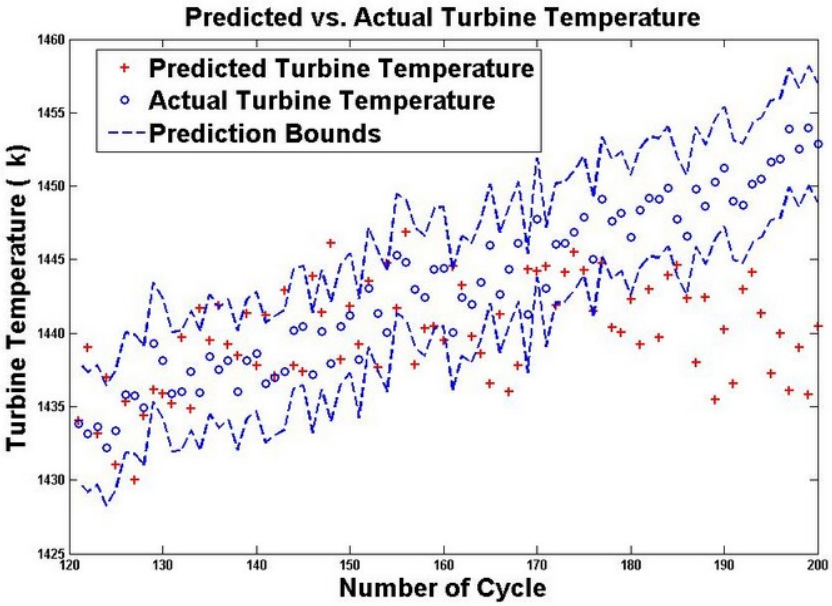

Figure 4.68: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$.

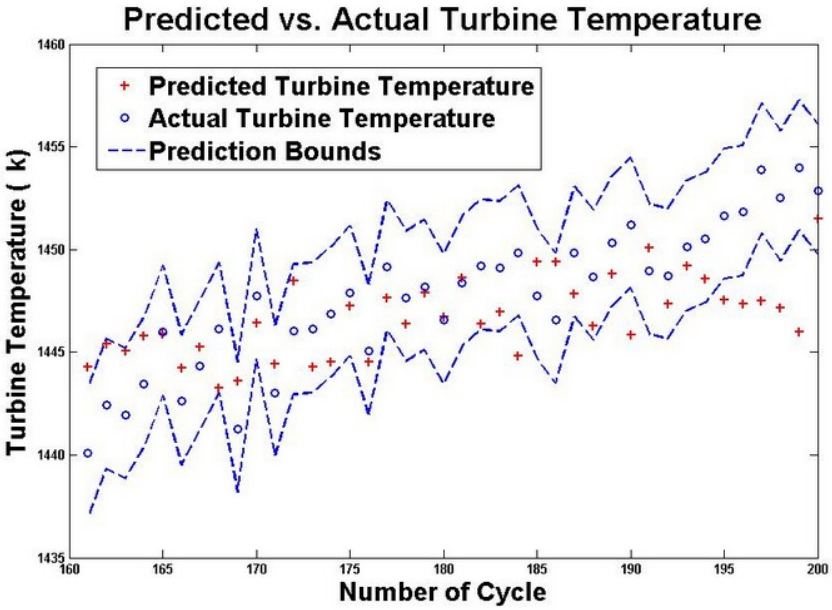

Figure 4.69: The 8 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$.

Table 4.66: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	6.6665	5.7236	8.7631
3	4.9613	5.2806	7.2216
4	**3.1573**	**5.7938**	**6.5664**
5	5.6156	4.7804	7.3554
6	6.3504	4.6937	7.8792
7	6.8274	6.5262	8.7619
8	7.6073	5.0773	9.1284

Table 4.67: An 8 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	3.3740	5.2178	6.1587
3	4.2172	4.3670	6.0315
4	4.4924	3.5524	5.6996
5	**3.0664**	**3.0065**	**4.2681**
6	3.5594	3.5779	5.0150
7	4.8131	4.1029	6.2912
8	5.8251	3.8360	6.9483

Section 2.5 and equations (2.6.1)-(2.6.4). It is assumed that this deterioration occurs in 200 simultaneous flights. Therefore, a total of 200 data points are available to train and test the networks. As done in Sections 4.1.3.1-4.1.3.3, the applicability of the Elman neural network to predict turbine temperature for multi-flights ahead is investigated using different numbers of training and testing data sets.

The turbine temperature is now predicted for 2 flights ahead. A total number of 80 data sets are used to train different Elman network architectures. The trained networks are then evaluated by using 120 data points, and the differences between the network's output and actual output are calculated. The statistical error measures are

illustrated in Table 4.68. The network with 6 hidden neurons has the best performance in predicting turbine temperature. The actual turbine temperature and predicted values for this network are depicted pointwise in Figure 4.70 along with the upper and the lower prediction bounds where 40% of the predicted points are within the bounds.

Table 4.68: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	8.0600	14.6631	16.6787
3	13.5524	8.1041	15.7733
4	11.5534	7.8173	13.9313
5	1.2491	12.9811	12.9871
6	**7.6364**	**6.8518**	**10.2407**
7	10.9578	7.4031	13.2069
8	10.0160	9.0986	13.5061

By increasing the number of training data points to 120, the lowest RMSE is achieved when the network has 4 hidden neurons. The summary of the prediction errors for 7 Elman network structures are shown in Table 4.69. Figure 4.71 shows the predicted values for the network 3-4-1 where 56.25% of the predicted values are within the prediction intervals.

Next the Elman networks are trained by using 160 data sets. The weights and biases are fixed and the networks are tested with 40 unseen data. The summary of the prediction errors are presented in Table 4.70. The Elman network 3-5-1 has the lowest RMSE equal to $3.5316K$. The mean and standard deviation are $2.7712K$ and $2.0170K$, respectively. The actual and predicted points are shown in Figure 4.72 where 77.5% of the predicted values are within the upper and the lower prediction intervals.

A 5 flights ahead turbine temperature is now predicted using different number of

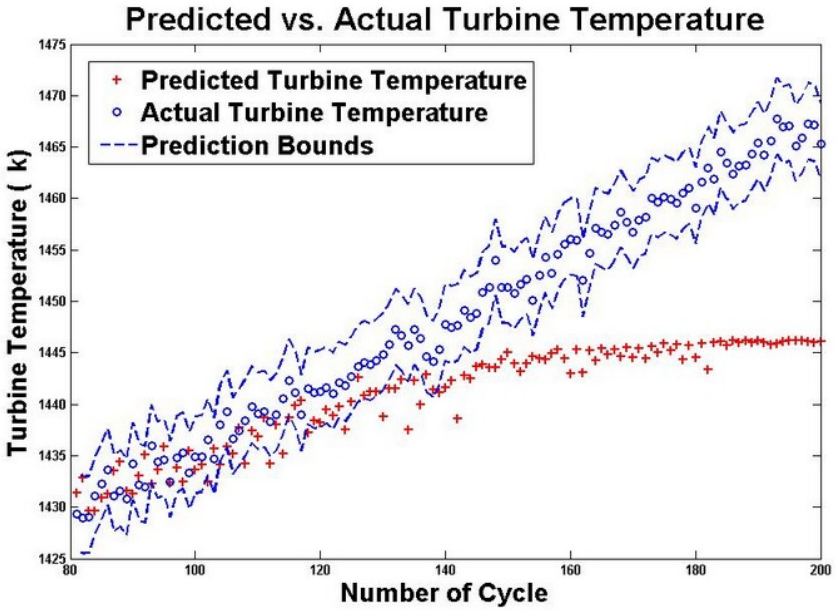

Figure 4.70: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-6-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$.

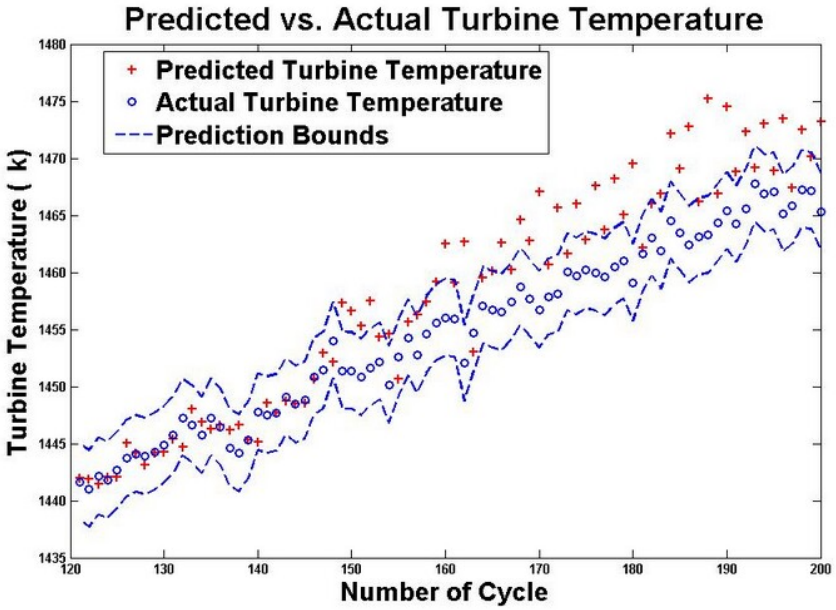

Figure 4.71: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$.

Table 4.69: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	10.1601	5.0045	11.3119
3	7.5663	5.3709	9.2593
4	**6.2317**	**5.0496**	**8.0008**
5	7.3853	4.8796	8.8349
6	7.5701	5.1710	9.1494
7	8.4786	5.7557	10.2274
8	8.0634	6.2049	10.1507

Table 4.70: A 2 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.1886	2.3905	4.8080
3	3.5808	2.0533	4.1150
4	3.2616	1.8031	3.7159
5	**2.7712**	**2.0170**	**3.5316**
6	3.1450	2.1893	3.8163
7	3.6321	2.6454	4.4739
8	3.9005	2.3839	4.5557

training and testing data points for different Elman network structures. In the first case 40% of the entire available data are used in the training phase and the networks are then tested with 120 data. The mean error of prediction, standard deviation and RMSE are summarized in Table 4.71. Next, the number of training data increases to 120 and the networks are tested with 80 data as shown in Table 4.72, and finally the prediction errors of the networks which are trained by using 160 data points are summarized in Table 4.73 where the network with 5 hidden neurons has the lowest RMSE.

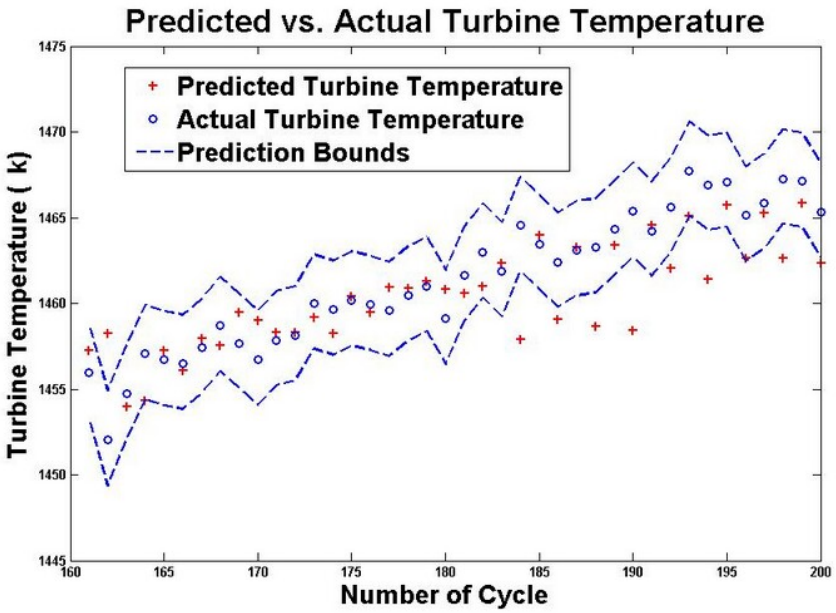

Figure 4.72: The 2 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$.

The predicted values for the optimal networks shown in Tables 4.71-4.73 are depicted in Figures 4.73-4.75. The actual and predicted data of the network 3-5-1 trained by using 40% of the entire data sets are shown pointwise in Figure 4.71 where 20.83% of the predicted turbine temperatures are within the prediction intervals. This value increases to 47.5% for the network 3-4-1 which is trained by using 60% of the available data, and finally 70% for the network 3-5-1 trained by using 80% of the entire data points.

4.1.3.5 Summary of the Results

The mean, standard deviation and RMSE of the turbine temperature prediction of the optimal networks found in Section 4.1.3 under presence of both compressor fouling and turbine erosion are summarized in Tables 4.74-4.77, where N_{train} is the number of data used in the training phase and N_{test} is the number of data which were used to

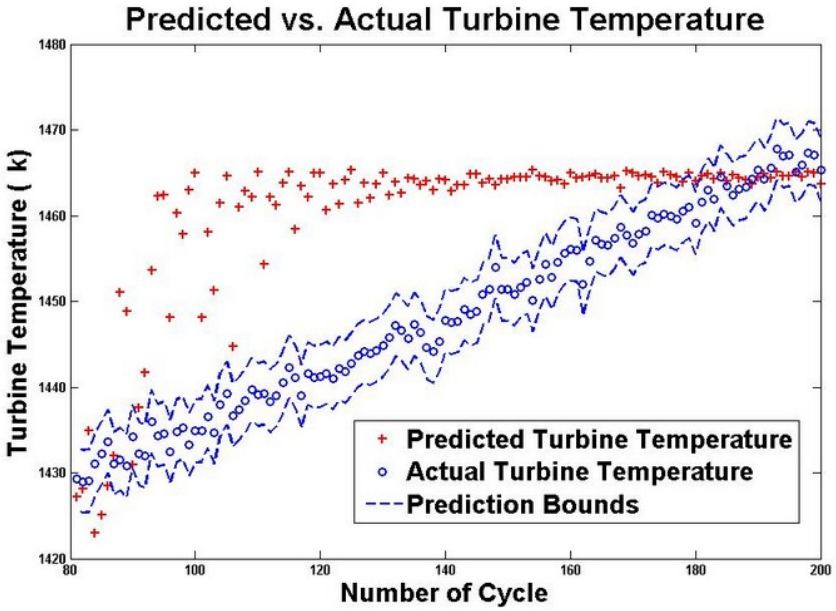

Figure 4.73: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$.

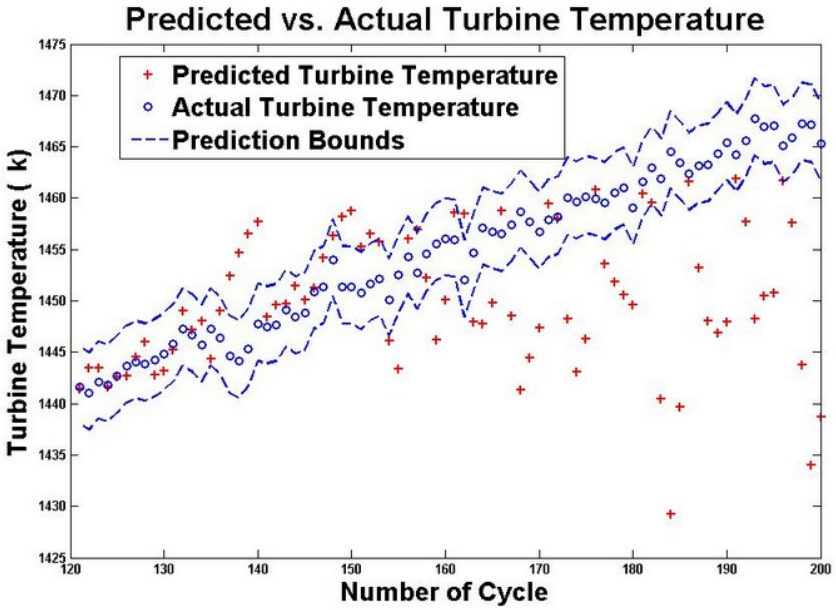

Figure 4.74: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-4-1 trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$.

Table 4.71: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 40% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	16.1962	10.3088	19.1756
3	14.5396	10.3822	17.8407
4	12.5910	9.3492	15.6593
5	**11.7348**	**9.3395**	**14.9735**
6	13.4131	9.5548	16.4452
7	14.9727	11.0984	18.6099
8	15.7119	11.3414	19.3499

Table 4.72: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 60% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	11.5078	8.9112	13.4014
3	9.0311	6.6780	11.2071
4	**4.7238**	**9.7767**	**10.8030**
5	11.4508	7.2715	13.5401
6	10.5464	9.0468	13.8581
7	13.7821	7.6639	15.7463
8	13.9011	7.8308	15.9310

test the trained network, and N_F is the number of flights ahead which the networks are used to predict the turbine temperature.

As shown in Tables 4.74-4.77, the reliability of the Elman neural networks is highly dependent on the amount of fouling and erosion in the engine. The network RMSE increases 19.07% when the network predicts 2 flights ahead in presence of 3% compressor fouling and turbine erosion instead of 1%. The network prediction error decreases as the number of training data points increases. Based on Table 4.77, the network RMSE decreases 65.5% if the training data points increases from 80 to 160 data in 2 flights ahead prediction. The accuracy of the prediction also increases if

Table 4.73: A 5 flight ahead turbine temperature prediction error for different number of hidden neurons trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

Number of hidden neurons	Mean (K)	Standard deviation (K)	RMSE (K)
2	4.4724	3.3408	5.5574
3	3.6597	3.1139	4.7799
4	3.7877	2.6983	4.6309
5	**3.6052**	**2.3468**	**4.2857**
6	3.3283	2.9702	4.4361
7	3.1297	3.7598	4.8557
8	4.4430	3.5645	5.6682

Table 4.74: Summary of the prediction errors for each scenario in presence of $FI = 1\%$ and $EI = 1\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-5-1	5.7737
120	80	2	3-5-1	5.1942
160	40	2	3-3-1	2.8579
80	120	5	3-4-1	6.6543
120	80	5	3-5-1	5.6227
160	40	5	3-5-1	3.3003
80	120	8	3-4-1	7.5050
120	80	8	3-5-1	6.2444
160	40	8	3-3-1	4.1819

the networks are trained by using more data points. Only 20.83% of the predicted values are within the upper and the lower prediction bounds when the network which is trained with 80 data sets predicts 5 flights ahead turbine temperature in presence of 3% fouling and erosion. However, this value increases to 70% if the network is trained by using 160 data sets as presented in Table 4.77.

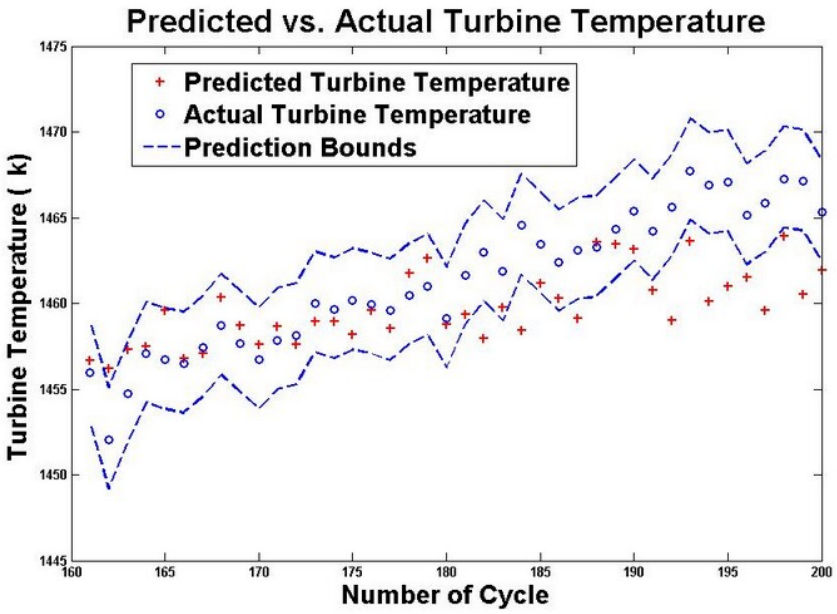

Figure 4.75: The 5 step ahead predicted/actual turbine temperature along with prediction intervals using the Elman 3-5-1 trained with 80% of the available data for $FI = 3\%$ and $EI = 3\%$.

4.2 Comparison of the NARX Neural Network and the Elman Neural Network

There should be a measure to compare the capability of the NARX and Elman neural networks. Using an appropriate neural network will increase the accuracy of prediction for maintenance actions. Model selection refers to the problem of using the data to select a model from a list of models [170]. Model selection should be based on the fact that it is impossible to find the "true" model that generates the data we observed. However, it should be based on a well-justified criterion to find the "best" model [168]. Model selection is a trade-off between bias (the distance between the average prediction and the actual value) and variance (spread of the prediction around the actual points). In other words, there is usually an improvement in the fit by increasing the parameters in the model, but at the same time parameter estimates

Table 4.75: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ and $EI = 2\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-6-1	8.7059
120	80	2	3-5-1	7.1789
160	40	2	3-4-1	3.2062
80	120	5	3-5-1	11.4291
120	80	5	3-5-1	8.7961
160	40	5	3-4-1	3.7578

Table 4.76: Summary of the prediction errors for each scenario in presence of $FI = 2\%$ and $EI = 3\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-5-1	6.2641
120	80	2	3-5-1	5.3468
160	40	2	3-3-1	3.0333
80	120	5	3-4-1	7.3380
120	80	5	3-3-1	6.0366
160	40	5	3-5-1	3.3688
80	120	8	3-6-1	10.2450
120	80	8	3-4-1	6.5664
160	40	8	3-5-1	4.2681

are worse because there is less data per parameter, and there is an increase in the computational time [169]. There are various model selection criteria in the literature; namely Akaike information criterion, Bayesian information criterion, deviance information criterion, etc [171].

Evaluation of the networks described previously are conducted by using Normalized Bayesian Information Criterion (NBIC) which has been widely used for model identification in time-series studies [172]. Therefore, suitable model can be found in each scenario for prediction. The NBIC can be defined as:

$$NBIC = ln(\sigma^2) + k\frac{ln(n)}{n} \tag{4.2.1}$$

Table 4.77: Summary of the prediction errors for each scenario in presence of $FI = 3\%$ and $EI = 3\%$ using Elman neural network.

N_{train}	N_{test}	N_F	Network structure	RMSE (K)
80	120	2	3-6-1	10.2407
120	80	2	3-4-1	8.0008
160	40	2	3-5-1	3.5316
80	120	5	3-5-1	14.9735
120	80	5	3-4-1	10.8030
160	40	5	3-5-1	4.2857

where σ^2 is the variance of the prediction error calculated from equation (4.2.2), k is the total number of parameters in the neural network, and n is the number of observations. It should be noted that smaller value for the NBIC implies that the model can predict the values better.

$$\sigma = (standard - deviation)^2 \tag{4.2.2}$$

The NBIC calculated for the neural networks considered in this thesis for each scenario are now compared together in Tables 4.78-4.85 for the NARX networks and Tables 4.86-4.93 for the Elman networks, where N_F is the number of flights ahead. In order to obtain the NBIC for the NARX network 7-6-1 tested by using 120 data points which predicts 2 flights ahead turbine temperature under presence of 1% compressor fouling, the standard deviation is 2.3338 and the variance is 5.4466 where n is 120 , and the factor k is the sum of the connections between the inputs and the hidden neurons which is 42 and the total number of inputs is 7 and the connections between the hidden neurons and the output which is 6. Therefore, the total number of parameters in the network is 55, and from equation (4.2.1) the NBIC becomes 3.8892.

For the Elman network 3-4-1 which predicts 2 flights ahead turbine temperature in presence of 1% fouling, there are 2 delays fed back from the hidden neurons to

the input layer. Therefore, there are ((2*4)+1=9) inputs to the network and the k parameter is equal to the sum of the 9 inputs, 36 connections between the inputs and the hidden neurons, and 4 connections between the hidden neurons and the output neuron which is equal to 49 parameters. The standard deviation is 2.5805 and the variance is 6.6589 and n is 120. Thus, the NBIC becomes 3.8508.

Table 4.78: NBIC values for each case in presence of $FI = 1\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-6-1	5.4466	120	55	3.8892
2	7-5-1	3.4062	80	47	3.8
2	7-6-1	2.9299	40	55	6.1471
5	7-7-1	8.159	120	63	4.6125
5	7-8-1	4.7276	80	71	5.4424
5	7-6-1	3.3764	40	55	6.289
8	7-8-1	7.8815	120	71	6.852
8	7-8-1	4.6475	80	71	5.4254
8	7-6-1	6.0851	40	55	6.878

Table 4.79: NBIC values for each case in presence of $FI = 3\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-8-1	18.98	120	71	5.776
2	7-9-1	9.9944	80	79	6.6292
2	7-8-1	4.2638	40	71	7.9979
5	7-9-1	28.8605	120	79	6.5142
5	7-6-1	9.4421	80	55	5.4579
5	7-9-1	3.2942	40	79	8.4777
8	7-10-1	47.0198	120	87	7.3215
8	7-7-1	15.3170	80	63	6.1798
8	7-7-1	12.0798	40	63	8.3015

Comparing the calculated NBIC in Tables 5.1-5.8 for different scenarios for the NARX neural network and Tables 5.9-5.16 for the same scenarios for the Elman network, one can conclude that the Elman network has lower NBIC which implies

Table 4.80: NBIC values for each case in presence of $EI = 1\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-6-1	4.6786	120	55	3.7373
2	7-8-1	2.6024	80	71	4.8455
2	7-6-1	2.2807	40	55	5.8967
5	7-9-1	5.0332	120	79	4.7678
5	7-8-1	3.1994	80	71	5.052
5	7-6-1	2.4124	40	55	5.9528
8	7-8-1	9.5011	120	71	5.084
8	7-5-1	4.9301	80	47	4.1697
8	7-7-1	3.5253	40	63	7.0699
12	7-10-1	4.7759	120	87	5.0345
12	7-8-1	5.3726	80	71	5.5704
12	7-7-1	3.0828	40	63	6.9358

that for the same degradation and the same training and testing data points, the Elman network outperforms the NARX network. This is mainly because the number of parameters k plays an important rule in the calculation of NBIC. The Elman network has a lower number of delays and hidden neurons. Thus, it can learn the trend of the degradations more efficiently and quicker.

Table 4.81: NBIC values for each case in presence of $EI = 3\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-11-1	13.8041	120	95	6.4150
2	7-8-1	10.1111	80	71	6.2027
2	7-8-1	3.8749	40	71	7.9023
5	7-9-1	11.4595	120	79	5.5906
5	7-6-1	11.9467	80	55	5.493
5	7-7-1	3.5838	40	63	7.0864
8	7-10-1	14.8248	120	87	6.1672
8	7-7-1	10.3638	80	63	5.7892
8	7-7-1	8.0105	40	63	7.8907

Table 4.82: NBIC values for each case in presence of $FI = 1\%$ and $EI = 1\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-9-1	23.2102	120	79	6.2963
2	7-9-1	6.2785	80	79	6.1644
2	7-8-1	3.6336	40	71	7.838
5	7-11-1	12.868	120	95	6.3448
5	7-8-1	11.975	80	71	6.3719
5	7-6-1	9.1664	40	55	7.2877
8	7-10-1	12.0881	120	87	5.9631
8	7-9-1	15.4158	80	79	7.0626
8	7-7-1	7.4261	40	63	7.815

Table 4.83: NBIC values for each case in presence of $FI = 3\%$ and $EI = 2\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-10-1	37.5475	120	87	7.0965
2	7-9-1	21.4036	80	79	7.3908
2	7-9-1	3.1301	40	79	8.4266
5	7-9-1	56.3010	120	79	7.1825
5	7-10-1	17.023	80	87	7.6
5	7-9-1	5.2822	40	79	8.9499

Table 4.84: NBIC values for each case in presence of $FI = 2\%$ and $EI = 3\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-8-1	29.8946	120	71	6.2303
2	7-10-1	15.9528	80	87	7.5351
2	7-10-1	5.5385	40	87	9.735
5	7-11-1	50.427	120	95	7.7106
5	7-10-1	17.3222	80	87	7.6174
5	7-10-1	8.5743	40	87	10.1721
8	7-12-1	62.0093	120	103	8.2365
8	7-10-1	33.8223	80	87	8.2866
8	7-10-1	9.0378	40	87	10.2247

Table 4.85: NBIC values for each case in presence of $FI = 3\%$ and $EI = 3\%$ for the NARX neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	7-11-1	64.7268	120	95	7.9603
2	7-12-1	26.6369	80	103	8.9241
2	7-9-1	5.5281	40	79	8.9953
5	7-12-1	133.3909	120	103	9.0025
5	7-10-1	35.7927	80	87	8.3432
5	7-11-1	9.5345	40	95	11.016

Table 4.86: NBIC values for each case in presence of $FI = 1\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-4-1	6.6589	120	49	3.8508
2	3-3-1	5.4934	80	31	3.4016
2	3-3-1	3.54	40	31	4.123
5	3-5-1	4.1861	120	71	4.2643
5	3-3-1	4.0594	80	31	3.099
5	3-4-1	4.5565	40	49	6.0354
8	3-4-1	8.3405	120	49	4.076
8	3-5-1	4.4449	80	71	5.3808
8	3-4-1	5.1243	40	49	6.1528

Table 4.87: NBIC values for each case in presence of $FI = 3\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-5-1	18.5304	120	71	5.752
2	3-5-1	6.3373	80	71	5.7355
2	3-3-1	3.0947	40	31	3.9886
5	3-4-1	29.1243	120	49	5.3265
5	3-5-1	4.7450	80	71	5.4461
5	3-4-1	6.8105	40	49	6.4373
8	3-4-1	60.3993	120	49	6.0558
8	3-5-1	13.7952	80	71	6.5134
8	3-5-1	4.1873	40	71	7.9798

Table 4.88: NBIC values for each case in presence of $EI = 1\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-4-1	4.2952	120	49	3.4124
2	3-3-1	3.4239	80	31	2.9288
2	3-3-1	2.1806	40	31	3.6385
5	3-3-1	5.5117	120	31	2.9436
5	3-4-1	3.3361	80	49	3.8888
5	3-3-1	3.5118	40	31	4.115
8	3-5-1	6.1871	120	71	4.6551
8	3-3-1	3.7912	80	31	3.0307
8	3-3-1	2.5278	40	31	3.7862
12	3-4-1	7.6712	120	49	3.9924
12	3-4-1	6.5147	80	49	4.5581
12	3-5-1	1.2060	40	71	6.7351

Table 4.89: NBIC values for each case in presence of $EI = 3\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-4-1	9.6553	120	49	4.2224
2	3-5-1	3.6283	80	71	5.1778
2	3-3-1	4.8929	40	31	4.4467
5	3-4-1	8.0946	120	49	4.0461
5	3-5-1	5.9102	80	71	5.6657
5	3-5-1	3.528	40	71	7.8085
8	3-4-1	12.9398	120	49	4.5152
8	3-5-1	10.1856	80	71	6.21
8	3-4-1	4.8898	40	49	6.106

Table 4.90: NBIC values for each case in presence of $FI = 1\%$ and $EI = 1\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-5-1	16.5421	120	71	5.6385
2	3-5-1	8.6847	80	71	6.0506
2	3-3-1	3.9398	40	31	4.23
5	3-4-1	27.9915	120	49	5.2868
5	3-5-1	11.1629	80	71	6.3016
5	3-5-1	1.2611	40	71	6.7797
8	3-4-1	32.8959	120	49	5.4482
8	3-5-1	22.668	80	71	7.01
8	3-3-1	9.3385	40	31	5.093

Table 4.91: NBIC values for each case in presence of $FI = 3\%$ and $EI = 2\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-6-1	21.9455	120	97	6.9584
2	3-5-1	28.0433	80	71	7.2228
2	3-4-1	3.8663	40	49	5.8712
5	3-5-1	67.2203	120	71	7.0406
5	3-5-1	26.471	80	71	7.1651
5	3-4-1	7.1786	40	49	6.4899

Table 4.92: NBIC values for each case in presence of $FI = 2\%$ and $EI = 3\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-5-1	22.4259	120	71	5.9428
2	3-5-1	11.6957	80	71	6.3482
2	3-3-1	7.9738	40	31	4.9351
5	3-4-1	42.8213	120	49	5.7119
5	3-3-1	22.9182	80	31	4.8299
5	3-5-1	6.8022	40	71	8.4651
8	3-6-1	48.1164	120	97	7.7435
8	3-4-1	33.5681	80	49	6.1976
8	3-5-1	9.0391	40	71	8.7493

Table 4.93: NBIC values for each case in presence of $FI = 3\%$ and $EI = 3\%$ for the Elman neural network.

N_F	Network structure	σ^2	n	k	NBIC
2	3-6-1	46.9472	120	97	7.7189
2	3-4-1	25.4984	80	49	5.9226
2	3-5-1	4.0682	40	71	7.9501
5	3-5-1	87.2263	120	71	7.3011
5	3-4-1	95.5839	80	49	7.2441
5	3-5-1	5.5075	40	71	8.2539

4.3 Conclusion

In this chapter, the turbine output temperature in an aircraft jet engine is predicted in presence of deteriorations. It is assumed that the engine has different compressor fouling and turbine erosion rates. A series of simulations are conducted to illustrate the effectiveness of the Elman neural network in turbine temperature prediction. This prediction leads to the choice of condition-based maintenance according to the data collected from the engine through continuous monitoring. A discussion on the simulation results for each scenario is also provided. The presented simulations show that the Elman network has the capability to learn the trend of the degradations successfully. After that, the NBIC is calculated to compare the predictability of the NARX neural networks and the Elman neural networks which shows that the Elman neural network is preferred to achieve a more reliable prognostics.

Chapter 5

Conclusions and Future Work

The objective of this thesis was to develop an intelligent-based approach for fault prognosis of aircraft engines. Towards this end, artificial neural networks were employed due to their great capability in learning the dynamics of non-linear systems and their capabilities to cope with the system complexity. The reliability of these networks are then evaluated to predict the turbine temperature for multi-flights ahead in presence of various deteriorations.

Two significant degradations which affect the performance of the engine namely, compressor fouling and turbine erosion were modelled. We have also considered the scenarios where both compressor fouling and turbine erosion occur at the same time. The thermodynamic parameters of the engine can be affected by these degradations which are modelled by a decrease in the compressor and turbine efficiency. A reduction in the compressor mass flow rate and an increase in the turbine mass flow rate are also associated with these degradations.

The prediction capabilities of two neural networks were compared. The first prediction scheme was achieved by using the non-linear autoregressive neural network with exogenous input (NARX). The structure of this network consists of three layers of input, hidden, and output layers with delays of the output fed back to the input

layer. The delays of the input are also given as an extra input to the network. The optimal networks were found for a specific degradation mode. The statistical prediction errors such as mean, standard deviation, and RMSE for each network were presented.

The second prediction scheme utilized the Elman neural network. The Elman network architecture is also composed of three layers of input, hidden, and output layers. The main difference between this network and the NARX network is that the delays from the output of the neurons in the hidden layer are fed back to the input layer. This network is used to predict the turbine output temperature for multi-flights ahead in presence of different degradation rates. These predicted values are compared to the actual turbine temperatures.

The capability of the NARX neural network and the Elman neural network is compared using the normalized Bayesian information criterion. The results show that for the same degradation and the same training and testing data points the Elman neural network outperforms the NARX neural network.

5.1 Suggestions for Future Work

A number of potential future direction of research to extend the current work can be investigated. Some of our plans for future research are explained in the following:

First, there are different degradations which affect the performance of the jet engine. However, in this thesis only the compressor fouling and the turbine erosion were modelled in our Simulink model. Deteriorations such as corrosion, hot temperature oxidation, thermal distortion in the combustion chamber, and the tip clearance are among the degradations which cause a change in the performance of the engine's components. Therefore, future techniques can be focused on the investigation of the engine's health in the case when these degradations occur in the engine.

Second, the delays associated with the NARX network and the Elman network were assumed to be fixed. One can analyze the effects of the number of delays in the performance of the networks. Moreover, adaptive networks where the delays themselves are being updated along with the weights of the networks can also be investigated to see if the prediction accuracy can be improved.

Third, other neural network approaches can also be applied to predict multi-flights ahead turbine temperature. The reliability of the dynamic networks such as time delayed neural networks (TDNN), radial basis function neural networks (RBF), and Jordan networks can also be examined through simulations. Comparison of these other architectures may help to identify the best structure for this type of problem.

Fourth, as another recommendation for future studies, other criterion can be applied to find the proper model for prediction such as Akaike information criterion, deviance information criterion, cross-validation, and stepwise regression.

Bibliography

[1] M. Naeem, R. Singh, and D. Probert, "Implications of Engine Deterioration for Fuel Usage," *Applied Energy,* vol. 59, no. 2-3, pp. 125-146, 1998.

[2] T. Brotherton, G. Jahns, J. Jacobs, and D. Wroblewski, "Prognosis of Faults in Gas Turbine Engines," *Proceedings of the IEEE Aerospace Conference,* New York, 2000.

[3] A. Heng, S. Zhang, A. C. C. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges, and opportunities," *Rotating machinery prognostics: State of the art, challenges, and opportunities,* vol. 23, no. 3, pp. 724-739, 2009.

[4] R. Kothamasu, S. H. Huang, W. H. VerDuin, "System health monitoring and prognostics- a review of current paradigms and practices," *The International Journal of Advanced Manufacturing Technology,* vol. 28, no. 9-10, pp. 1012-1024, 15 March 2006.

[5] H. C. Pusey, "Turbomachinery Condition Monitoring and Failure Prognosis," *SAVIAC/Hi-Test Laboratories,* Winchester, Virginia, March 2007.

[6] J. Endrenyi, S. Aboresheid, R. N. Allan, "The Present Status of Maintenance Strategies and the Impact of Maintenance on Reliability," *IEEE Transactions on Power Systems,* vol. 16, no. 4, pp. 638-646, November 2001.

[7] A. K. S. Jardine, D. Lin, D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing,* vol. 20, no. 7, pp. 1483-1510, 2006.

[8] L. R. Contreras, C. Modi, A. Pennathur, "Integrating Simulation Modelling and Equipment Condition Diagnostics for Predictive Maintenance Strategies -A Case Study," *Winier Simulation Conference,* 2002.

[9] P. C. Escher, "Pythia: An Object-Oriented Gas Path Analysis Computer Program for General Applications," Cranfield University, Cranfield, October 1995.

[10] J. Luo, K. Pattipati, L. Qiao, and S. Chigusa, "Model-based prognostic techniques applied to a suspension system," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 38, no. 5, pp. 1156-1168, 2008.

[11] S. Azirah Asmai, B. Hussin, M. Mohd Yusof, "A Framework of an Intelligent Maintenance Prognosis Tool," *Second International Conference on Computer Research and Development,* IEEE, pp. 241-245, 2010.

[12] A. Babbar, E. M. Ortiz, V. L. Syrmos, M. M. Arita, "Advanced Diagnostics and Prognostics for Engine Health Monitoring," *Aerospace Conference,* IEEE, pp. 1-10, 2009.

[13] M. Zedda, R. Singh, "Gas Turbine Engine and Sensor Fault Diagnosis Using Optimization Techniques," *Proceedings of the AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit,* Los Angeles, California, 1999.

[14] Y. G. Li, "Performance Analysis Based Gas Turbine Diagnostics: A Review," *Journal of Power and Energy,* vol. 216, no. 5, p. 363-377, 2002.

[15] S. O. T. Ogaji, S. Sampath, R. Singh, S. D. Probert, "Parameter selection for diagnosing a gas-turbines performance-deterioration," *Applied Energy,* vol. 73, no.1, pp. 25-46, 2002.

[16] L. A. Urban, "Parameter Selection for Multiple Fault Diagnostics of Gas Turbine Engines," *Journal of Engineering for Power,* vol. 97, no. 2, 1974.

[17] P. Fuster, A. Ligeza, and J. Aguilar Martin, "Abductive Diagnostic Procedure Based on an AND/OR/NOT Graph for Expected Behaviour: Application to a Gas Turbine," *10th International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management,* Espo, Finland, 1997.

[18] M. Zedda, and R. Singh, "Gas turbine engine and sensor fault diagnosis using optimization techniques," *Journal of Propulsion and Power,* vol. 18, no. 5, pp. 1019-1025 , 2002.

[19] Y. Zhang, J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Review in Control,* vol. 32, no. 2, pp. 229-252, 2008.

[20] S. Simani, R. Patton, and C. Fantuzzi, "Model-Based Fault Diagnosis in Dynamic Systems Using Identification Techniques," New York: Springer-Verlag, 2003.

[21] R. Beard, "Failure accommodation in linear systems through self reorganization," *Ph.D. thesis,* Massachusets Institute of Technology, Mass, USA, 1971.

[22] A. S. Willsky and H. L. Jones, "Generalized likelihood ratio approach to detection and estimation of jumps in linear systems," *IEEE Transactions on Automatic Control,* vol. 21, no. 1, pp. 108-112, 1976.

[23] I. Hwang, S. Kim, Y. Kim and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Transactions on Control Systems Technology,* vol. 18, no. 3, pp. 636-653, 2010.

[24] L. Marinai, D. Probert, R. Singh, "Prospects for aero gas-turbine diagnostics: a review," *Applied Energy,* vol. 79, no. 1, pp. 109-126, 2004.

[25] G. Denny, "F-16 Jet Engine Trending and Diagnosis with Neural Networks," *Application of Neural Networks,* vol. 4, no. 4, pp. 419-422, 1965.

[26] N. Meskin, E. Naderi, and K. Khorasani, "A Multiple Model-Based Approach for Fault Diagnosis of Jet Engines," *IEEE Transactions on Control Systems Technology,* vol. 21, no. 1, pp. 254-262, January 2013.

[27] R. K. Yedavalli, P. Shankar, M. Siddiqi, "Modelling, Diagnostics, And prognostics of A Two-Spool Turbofan Engine," *American Institute of Aeronautica and Astronautics,* Ohio State University Columbus, 2005.

[28] A. Valdes, K. Khorasani, and L. Ma, "Dynamic neural network-based fault detection and isolation for thrusters in formation flying of satellites," *Proc. of ISNN 2009, 6th International Symposium on Neural Networks: Advances in Neural Networks - Part III,* Springer-Verlag, 2009.

[29] R. Mohammadi, E. Naderi, K. Khorasani, and S. Hashtrudi-Zad, "Fault diagnosis of gas turbine engines by using dynamic neural networks," *ASME Turbo Expo,*Glasgow, UK, 2010.

[30] R. Joly, S. Ogaji, R. Singh, and S. Probert, "Gas turbine diagnostics using artificial neural-networks for a high bypass ratio military turbofan engine," *Applied Energy,* vol. 78, no. 4, pp. 397418, 2005.

[31] S. Sarkar, X. Jin, A. Ray, "Data-Driven Fault Detection in Aircraft Engines With Noisy Sensor Measurements," *Journal of Engineering for Gas Turbines and Power,* vol. 133, no.8, 2011.

[32] S. Sarkar, C. Rao, and A. Ray, "Statistical estimation of multiple faults in aircraft gas turbine engines," *Aerospace Engineering,* 2009.

[33] E. Chu and D. Gorinevsky, "Detecting Aircraft Performance Anomalies from Cruise Flight Data," *Aerospace Conference,* Atlanta, GA, 2010.

[34] C. Nan, F. Khan, M. T. Iqbal, "Real-time fault diagnosis using knowledge-based expert system," *Process Safety and Environmental Protection,* vol. 86, no. 1, pp. 55-71, 2008.

[35] K. Patan, and T. Parisini, "Identification of neural dynamic models for fault detection and isolation: the case of a real sugar evaporation process," *Journal of Process Control,* vol. 15, no. 1, pp. 67-79, 2005.

[36] I. A. Al-Zyoud, "Neural network-based actuator fault detection and isolation for the attitude control subsystem of a satellite," *Master's thesis,* Concordia University, 2005.

[37] L. Li, L. Ma, and K. Khorasani, "A dynamic recurrent neural network fault diagnosis and isolation architecture for satellite's actuator/thruster failures," *Advances in Neural NetworksISNN 2005,* pp. 574-583. Springer Berlin Heidelberg, 2005.

[38] J. Lee, J. Ni, and D. Djurdjanovic, H. Qiu, and H. Liao, "Intelligent prognostics tools and e-maintenance," *Computers in industry,* vol. 54, no. 1, pp. 476-489, 2006.

[39] F. L. Greitzer, R. A. Pawlowski, "Embedded Prognostics Health Monitoring," *International Instrumentation Symposium, Embedded Health Monitoring Workshop,* May 2002.

[40] A. Ray, S. Tangirala, "Stochastic Modeling of Fatigue Crack Dynamics for On-Line Failure Prognostics," *IEEE Transactions on Control Systems Technology,* vol. 4, no. 4, pp. 443-451, July 1996.

[41] S. A. Asmai, B. Hussin, M. M. Yusof, "A Framework of an Intelligent Maintenance Prognosis Tool," *Second International Conference on Computer Research and Development,* IEEE, 2010.

[42] M. Orchard, B. Wu, and G. Vachtsevanos, "A Particle Filtering Framework for Failure Prognosis," *World Tribology Congress,* Washington, D.C., USA, 2005.

[43] M. Schwabacher and K. Goebel, "A Survey of Artificial Intelligence for Prognostics," *NASA Ames Research Center,* Moffett Field, CA.

[44] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, B. Wu, "Intelligent Fault Diagnosis and Prognosis for Engineering Systems," *Wiley,* 2006.

[45] M. E. Orchard, G. J. Vachtsevanos, "A Particle Filtering-based Framework for Real-time Fault Diagnosis and Failure Prognosis in a Turbine Engine," *15th Mediterranean Conference on Control & Automation,* Athens- Greece, July 27-29, 2007.

[46] M. Abbas, A. Ferri, M. Orchard and G. Vachtsevanos, "An intelligent diagnostic/prognostic framework for automotive electrical systems," *IEEE Intelligent Vehicles Symposium,* Istanbul, Turkey, 2007.

[47] D. He and E. Bechhoefer, "Probabilistic Model Based Algorithms for Prognostics," *Aerospace Conference,* IEEE, 2006.

[48] M. Watson, C. Byington, D. Edwards, S. Amin, "Dynamic modelling and wear-based remaining useful life prediction of high power clutch systems," *Tribology Transactions,* vol. 48, no. 2, pp. 208-217, 2005.

[49] D. Chelidze, "Multimode damage tracking and failure prognosis in electromechanical system," *SPIE Conference Proceedings,* pp. 1-12, 2002.

[50] M. E. Orchard, "A Particle Filtering-Based Framework for On-line Fault Diagnosis and Failure Prognosis," *ph.D. thesis,* Georgia Institute of Technology, 2007.

[51] O. E. Dragomir, R. Gouriveau, and F. Dragomir, "Review of Prognostic Problem in Condition-Based Maintenance," *European Control Conference,* Budapest, Hungary, 2009.

[52] W. Q. Wang, M. F. Golnaraghi, F. Ismail "Prognosis of machine health condition using neuro-fuzzy systems," *Mechanical Systems and Signal Processing,* vol. 18, no. 4, pp. 813–831, 2004.

[53] V. T. Tran, B. Yang, M. Oh, A. Chit Chiow Tan, "Machine condition prognosis based on regression trees and one-step-ahead prediction," *Mechanical Systems and Signal Processing,* vol. 22, no. 5, pp. 11791193, 2008.

[54] R. B. Chinnam and P. Baruah, "Autonomous diagnostics and prognostics through competitive learning given HMM-based clustering," *Proceedings of the International Joint Conference on Neural Networks,* vol. 4, pp. 2466-2471, 2003.

[55] A. K. Garga, K. T. Meclmtic, R. L. Campbell, C. C. Vang and M. S. Lebolil, "Hybrid reasoning for prognostic learning in cbm systems," *Proceedings of IEEE Aerospace Conference,* vol. 6, pp. 2957-2969, 2001.

[56] L. Datong, P. Yu, P. Xiyuan, "Online Adaptive Status Prediction Strategy for Data-Driven Fault Prognostics of Complex Systems," *Prognostics and System Health Management Conference,* Shenzhen, 2011.

[57] Y. L. Dong, Y. J. Gu, K. Yang and W. K. Zhang, "A combining condition prediction model and its application in power plant," *Proceedings of 2004 International Conference on Machine Learning and Cybernetics,* vol. 6, pp. 3474-3478, 2004.

[58] G. Vachtsevanos, P. Wang, "Fault Prognosis Using Dynamic Wavelet Neural Network," *AUTOTESTCON Proceedings, IEEE Systems Readiness Technology Conference,* 2001.

[59] J. Yan, M. Koc and J. Lee, "A prognostic algorithm for machine performance assessment and its application," *Production Planning and Control,* vol. 15, no. 8, pp. 796-801, 2004.

[60] C. Kwan, X. Zhang, R. Xu, and L. Haynes, "A novel approach to fault diagnostics and prognostics," *Proceedings of the 2003 IEEE International Conference on Robotics and Automation,* Vols. 1-3, pp. 604-609, 2003.

[61] R. Jardim-Goncalves, M. Martins-Barata, J. Alvaro Assis-Lopes, A. Steiger-Garcao, "Application of Stochastic Modelling to Support Predictive Maintenance for Industrial Environments," *IEEE International Conference on Systems, Man, and Cybernetics,* vol. 1, pp. 117-122, 1996.

[62] U. Thissen, R. van Brakel, A. P. de Weijer, W. J. Melssen, L. M. C. Buydens, "Using support vector machines for time series prediction," *Chemometrics and Intelligent Laboratory Systems,* vol. 69, no. 1, pp. 35-49, 2003.

[63] B. E. Parker, T. M. Nigro, M. P. Carley, R. L. Barron, D. G. Ward, H. V. Poor, D. Rock, and T. A. DuBois, "Helicopter gearbox diagnostics and prognostics using vibration signature analysis," *Proceedings of the SPIE- The International Society for Optical Engineering,* vol. 1965, pp. 531-542, 1993.

[64] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, J. Lee, "Implications of Engine Deterioration for Fuel Usage," *Applied Energy,* vol. 59, no. 2, pp. 125-146, 1998.

[65] C. Frelicot, "A fuzzy-based prognostic adaptive system," *Journal Europeen des Systemes Automatises,* vol. 30, no. 2, pp. 167-176, 1996.

[66] P. P. Bonissone, "Domain Knowledge and Decision Time: A Framework for Soft Computing Applications," *International Symposium on Evolving Fuzzy Systems,* September 2006.

[67] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing,* vol. 50, no. 6, pp. 159-175, 2003.

[68] F. Zhao, Z. Tian, and Y. Zeng, "Uncertainty Quantification in Gear Remaining Useful Life Prediction Through an Integrated Prognostics Method," *IEEE Transactions on Reliability,* vol. 62, no. 1, pp. 146-159, 2013.

[69] Y. Gao, M. J. Er, "NARMAX time series model prediction: feedforward and recurrent fuzzy neural network approaches," *Fuzzy Sets and Systems,* vol. 150, no. 2, pp. 331-350, 2005.

[70] B. Satish, N. D. R. Sarma, "A fuzzy BP Approach for Diagnosis and Prognosis of Bearing faults in Induction Motors," *Power Engineering Society General Meeting,* pp. 2291-2294, IEEE, 2005.

[71] L. Atlas, G. Bloor, T. Brotherton, L. Howard , L. Jaw, G. Kacprzynski, G. Karsai, R. Mackey, "An Evolvable Tri-Reasoner IVHM System," *Proceedings of the IEEE Aerospace Conference,* New York, 2001.

[72] F. Harjes, B. Scholz-Reiter, A. Kaviani Mehr, "Elman Networks for the Prediction of Inventory Levels and Capacity Utilization," *International Journal of Applied Mathematics and Informatics,* vol. 5, no. 4, pp. 283-290, 2011.

[73] K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Transactions on Neural Networks,* vol. 1, no. 1, pp. 4-27, 1990.

[74] J. Lee, "A systematic approach for developing and deploying advanced prognostics technologes and tools: methodology and applications," *Proceedings of the 2th World Congress on Engineering Asset Management,* Harrogate, Uk, 2007.

[75] P. Tse, D. Atherton, "Prediction of machine deterioration using vibration based fault trends and recurrent neural networks," *Transactions of ASME: Journal of Vibration and Acoustics,* vol. 121, no. 3, pp. 355-362, 1999.

[76] R. C. M. Yam, P. W. Tse, L. Li, P. Tu, "Intelligent predictive decision support system for CBM," *The International Journal of Advanced Manufacturing Technology,* vol. 17, no. 5, pp. 383-391, 2001.

[77] S. Jianzhong, Z. Hongfu, Y. Haibin, and M. Pecht, "Study of Ensemble Learning-Based Fusion Prognostics," *Prognostics and System Health Management Conference,* Macau, 2010.

[78] G. Xue, L. Xiao, M. Bie, S. Lu, "Fault prediction of boilers with fuzzy mathematics and RBF neural network," *Communications, Circuits and Systems,* vol. 2, pp. 1012-1016, IEEE, May 2005.

[79] N. Z. Gebraeel, and M. A. Lawley, "A Neural Network Degradation Model for Computing and Updating Residual Life Distributions," *IEEE Transactions on Automation Science and Engineering,* vol. 5, no. 1, pp. 154-163, 2008.

[80] O. Dragomir, R. Gouriveau, and N. Zerhouni, "Adaptive Neuro-Fuzzy Inference System for mid term prognosis error stabilization," *International Journal of Computers, Communications & Control,* vol. 1, p. 6, 2008.

[81] R. Zemouri, "Recurrent Radial Basis Function network for time-series prediction," *Engineering Application of Artificial Intelligence,* vol. 16, no. 5, pp. 453-463, 2003.

[82] S. Zhang, and R. Ganesan, "Multivariable trend analysis using neural networks for intelligent diagnostics of rotating machinery," *Transactions of the ASME Journal of Engineering for Gas Turbines and Power,* vol. 119, no. 2, pp. 378-384, 1997.

[83] S. Sina Tayarani Bathaie, Z. Sadough, and K. Khorasani, "Fault Detection of Gas Turbine Engines using Dynamic Neural Networks," *IEEE Canadian Conference on Electrical and Computer Engineering,* Montreal, Quebec, 2012.

[84] K. Patan, "Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes," *Springer, ISBN 978-3-540-79871-2,* February, 2008.

[85] A. Yazdizadeh, K. Khorasani, "Adaptive time delay neural network structures for non-linear system identification," *Neurocomputing,* vol. 47, no. 1, pp. 207-240, 2002.

[86] P. Wang, F. Golnaraghi, and F. Ismail, "A robust prognostic system for real time industrial applications," *4th International Conference on Industrial Automation,* 2003.

[87] M. Ishikawa, T. Moriyama, "Prediction of time series by a structural learning of neural networks," *Fuzzy Sets and Systems,* vol. 82, no. 2, pp. 167-176, 1996.

[88] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks,* vol. 1, no. 1, pp. 4-27, 1990.

[89] S. Z. Qin, H. T. Su, and T. J. McAvoy, "Comparison of four neural net learning methods for dynamic system identification," *IEEE Transactions on Neural Networks,* vol. 3, no. 1, pp. 122-130, 1992.

[90] H. T. Su, T. J. McAvoy, and P. Werbos, "Long-term predictions of chemical processes using recurrent neural networks: A parallel training approach," *Industrial Engineering and Chemical Research,* pp. 1338-1352, 1992.

[91] S. Chen, S. A. Billings, and P. M. Grant, "Non-linear system identification using neural networks," *International Journal of Control,* vol. 51, no. 6, pp. 1191-1214, 1990.

[92] P. Amani, A. Robertsson, "NARX-based Multi-step Ahead Response Time Prediction for Database Servers," *Intelligent Systems Design and Applications,* IEEE, 2011.

[93] E. Diaconescu, "The use of NARX Neural Network to predict Chaotic Time Series," *WSEAS Transactions on Computer Research,* vol. 3, no. 3, pp. 182-191, 2008.

[94] H. Y. Shen and L. C. Chang, "On-line multistep-ahead inundation depth forecasts by recurrent NARX networks," *Hydrology and Earth System Sciences Discussions,* vol. 9, no. 10, pp. 11999-12028, 2012.

[95] Y. Bar-Yam, "Dynamics of Complex Systems (Studies in Nonlinearity)," *Westview Press,* 2003.

[96] L. Leontaritis, S. Billings, "Input-output parametric models for non-linear systems," *International Journal of Control,* vol. 41, no. 2, pp. 303-328, 1985.

[97] C. Jiang, F. Song, "Sunspot forecasting by using chaotic time-series analysis and NARX network," *Journal of Computers,* vol. 6, no. 7, pp. 1424-1429, 2011.

[98] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P. Glorennec, H. Hjalmarsson, A. Juditsky, "Non-linear black-box modelling in system identification: a unified overview," *Automatica,* vol. 31, no. 12, pp. 1691-1724, 1995.

[99] S. Haykin, "Neural Networks and Learning Machines (3rd edition),"New Jersey, USA: Prentice Hall, 2008.

[100] De Jess, O. and M. Hagan, "Backpropagation Algorithms for a Broad Class of Dynamic Networks," *IEEE Transactions on Neural Networks,* vol. 18, no. 1, 2007.

[101] A. Suratgar, M. Tavakoli, and A. Hoseinabadi, "Modified Levenberg-Marquardt Method for Neural Networks Training," *World Academy of Science, Engineering and Technology,* 2005.

[102] T. Lin, C. L. Giles, B. G. Horne, S. Y. Kung, "A Delay Damage Model Selection Algorithm for NARX Neural Networks," *IEEE Transactions on Signal Processing ""Special Issue on Neural Networks",* vol. 45, no. 11, pp. 2719-2730, 1997.

[103] B. G. Horne and C. L. Gilles, "An experimental comparison of recurrent neural networks," *Neural Information Processing Systems,* vol. 7, pp. 697-704, 1995.

[104] Y. Gao, M. J. Er, "NARMAX time series model prediction: feedforward and recurrent fuzzy neural network approaches," *Fuzzy Sets and Systems,* vol. 150, no. 2, pp. 331-350, 2005.

[105] T. Lin, B. G. Horne, P. Tino, C. L. Giles, "Learning long-term dependencies in NARX recurrent neural networks," *IEEE Transactions on Neural Networks,* vol. 7, no. 6, pp. 1329-1351, 1996.

[106] D. C. Montgomery and E. A. Peek, "Introduction to Linear Regression Analysis, New York: Wiley, 1982.

[107] H. Xie, H. Tang, Y. Liao, "Time series prediction based on NARX neural networks: An advanced approach," *Proceedings of 8th International Conference on Machine Learning and Cybernetics,* Baoding, 2009.

[108] J. M. P. Menezes Jr., G. A. Barreto, "Long-term time series prediction with the NARX network: An empirical evaluation," *Neurocomputing,* vol. 71, no. 16, pp. 3335- 3343, 2008.

[109] H. T. Siegelmann, B. G. Horne,and and C. L. Giles "Computational Capabilities of Recurrent NARX Neural Networks," *IEEE*

Transactions On Systems, Man, and Cybernetics, vol. 27, no. 2, pp. 208-215, April 1997.

[110] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol. 37, no. 3, pp. 328-339, 1989.

[111] U. Thissena, R. van Brakela, A. P. de Weijerb, W. J. Melssena, L. M. C. Buydensa, "Using support vector machines for time series prediction," *Chemometrics and Intelligent Laboratory Systems,* vol. 69, no. 1, pp. 35-49, 2003.

[112] B. Yang, X. Mu, Q. Zhang, "Elman Neural Network based Temperature Prediction in Cement Rotary Kiln Calcining Process," *2010 International Conference on Intelligent Systems and Knowledge Engineering,* IEEE, 2010.

[113] C. Liu, D. Jiang and M. Zhao, "Application of RBF and Elman Neural Networks on Condition Prediction in CBM," *The Sixth International Symposium on Neural Networks,* pp. 847-855, Springer-Verlag Berlin Heidelberg 2009.

[114] J. Zhang, H. Chung, and W. Lo, "Chaotic Time Series Prediction Using a Neuro-Fuzzy System with Time-Delay Coordinates," *IEEE transactions on Knowledge and Data Engineering,* vol. 20, no. 7, pp. 956-964, 2008.

[115] F. M. Tseng, H. C. Yu, G. H. Tzeng, "Combining neural network model with seasonal time series ARIMA model," *Technological Forecasting & Social Change,* vol. 69, no. 1, pp. 71-87, 2002.

[116] C. J. Lin, C. H. Chen, "Identification and Prediction using recurrent compensatory neuro-fuzzy systems," *Fuzzy Sets and Systems,* vol. 150, no. 2, pp. 307-330, 2005.

[117] J. Liu, W. Wang, F. Golnaraghi, "A multi-step predictor with a variable input pattern for system state forecasting," *Mechanical Systems and Signal Processing,* vol. 23, no. 5, pp. 1586-1599, 2009.

[118] H. T. Pham, V. T. Tran, B. S. Yang, "A hybrid of non-linear autoregressive model with exogenous input and autoregressive moving average model for long-term machine state forecasting," *Expert Systems with Applications,* vol. 37, no. 4, pp. 3310-3317, 2010.

[119] J. S Armstrong, "Identification of Asymmetric Prediction Intervals through Causal Forces," *Journal of Forecasting,* vol. 20, no. 4, pp. 273-283, 2001.

[120] F. Meng, and W. Q. Meeker "Coverage Properties of Weibull Prediction Interval Procedures to Contain a Future Number of Failures," Iowa State University.Ames, IA 50011, 2011.

[121] NIST/SEMATECH "e-Handbook of Statistical Methods", http://www.itl.nist.gov/div898/handbook/April, 2012.

[122] J. Sterne,and B. Kirkwood "Essential medical statistics," Oxford: Blackwell Science. ISBN 0-86542-871-9.

[123] E. Naderi, N. Meskin, K. Khorasani, "Nonlinear Fault Diagnosis of Jet Engines by Using a Multiple Model Based Approach," *Journal of Engineering for Gas Turbines and Power,* vol. 134, no. 1, January 2012.

[124] 2012. [Online]. Available: http://www.aerospaceweb.org/question/conspiracy/q0265.shtml. [Accessed 21 05 2013].

[125] N. Meskin, E. Naderi, and K. Khorasani, "Fault Diagnosis of Jet Engines by using a Multiple Model-based Approach,"ASME, 2010.

[126] H. Saravanamuttoo, G. Rogers, and H. Cohen, "Gas Turbine Theory," *ISBN-13:978-0130158475,* Pearson Education, 2001.

[127] G. Merrington, "Fault Diagnosis in Gas Turbines Using a Model Based Technique," *Journal of Engineering for Gas Turbine and Power,* vol. 116, no. 2, pp. 374-380, 1994.

[128] S. S. Tayarani-Bathaie, Z. N. Sadough Vanini, K. Khorasani, "Dynamic neural network-based fault diagnosis of gas turbine engines," *Neurocomputing,* 2013.

[129] W. P. J. Visser and M. J. Broomhead, "GSP: A generic object-oriented gas turbine simulation environment," *National Aerospace Laboratory (NLR),* 2000.

[130] G. D. Team, "GSP 11 User Manual," *National Aerospace Laboratory,* 2011.

[131] R. Kurz, K. Brun, "Degradation in Gas Turbine Systems," *International Gas Turbine & Aeroengine Congress & Exhibition,* Munich, Germany, 2000.

[132] S. M. Flesland, "Gas Turbine Optimum Operation," *Master of Science in Product Design and Manufacturing,* Norwegian University of Science and Technology, December 2010.

[133] I. S. Diakunchak, "Performance deteriorations in industrial gas-turbines," *ASME Journal of Engineering for Gas-turbines and Power,* vol. 114, no. 2, pp. 161-168, 1992.

[134] G. F. Aker, and H. I. H. Saravanamuttoo, "Predicting Gas Turbine Performance Degradation Due to Compressor Fouling Using Computer Simulation Techniques," *Journal of Engineering Gas Turbines Power,* vol. 111, no. 2, pp. 343-400, 1989.

[135] "GE Measurement and Control,"[Online]. Available: http://www.ge-mcs.com/en/ndt-corrosion-inspection/turbine-blade-erosion.html. [Accessed 24 05 2013].

[136] A. Salar, "Development and Simulation of a Condition Monitoring and Fault Diagnosis System for Siemens V94.2 Gas Turbine using Kalman Filter Method," *Master of Engineering, K.N.Toosi University of Technology,* Tehran, Iran, February 2011.

[137] P. Zhu, and H. I. H. Saravanamutto, "Simulation of an Advanced Twin-Spool Industrial Gas Turbine," *ASME, Journal of Engineering Gas Turbines Power,* vol. 114, no. 2, pp. 180-185, April 1992.

[138] R. Kurz, K. Brun, "Gas Turbine Tutorial- Maintenance and Operating Practices Effects," *Turbomachinery Symposium,* 2007.

[139] M. Grewal, "Gas turbine engine's performance deterioration modelling and analysis," *PhD Thesis, Cranfield University,* 1988.

[140] J. D. MacLeod, V. Taylor and J. C. G. Laflamme, "Implanted Component Faults and Their Effects on Gas Turbine Engine Performance," *Journal of Engineering Gas Turbines Power,* vol. 114, no. 2, pp. 174-179, 1992.

[141] B. D. MacIsaac, "Engine performance and health monitoring models using steady-state and transient prediction methods," *Advisory Group for Aerospace Research and Development,* AGARD-LS-183,9.1-9.21, 1992.

[142] M. Naeem, R. Singh, D. Probert, "Implications of engine deterioration for creep life," *Applied Energy,* vol. 60, no. 4, pp. 183-223, 1998.

[143] A. Zwebek, "Combined Cycle Performance Deterioration Analysis," *PhD Thesis, Cranfield University,* 2002.

[144] C. B. Meher- Homji, A. Bromley, "Gas Turbine Axial Compressor Fouling and Washing," *Turbomachinery Symposium,* 2004.

[145] Z. Mustafa, "Analysis of Droplets in Compressor Gas Turbines," *PhD Thesis, Cranfield University,* UK, 2006.

[146] A. Tarabrin, V. Schurovsky, A. Bodrov, and J. Stalder, "Influence of axial compressor fouling on gas turbine unit perfomance

based on different schemes and with different initial parameters," *International Gas Turbine & Aeroengine Congress & Exhibition,* 1998.

[147] S. Gulen, P. Griffin, and S. Paolucci, "Real-time on-line performance diagnostics of heavy-duty industrial gas turbine," *Journal of Engineering for Gas Turbines and Power,* vol. 124, no. 4, pp. 910-921, 2002.

[148] P. Wilkinson, and L. K. Shark, "Automatic monitoring of gas turbine air intakes using color imaging techniques," *Non-Destructive Testing and Condition Monitoring,* vol. 46, no. 2, pp. 94-97, 2004.

[149] D. Bouris, R. Kubo, H. Hirata, and Y. Nakata, "Numerical comparative study of compressor rotor and stator blade deposition rates," *Journal of Engineering for Gas Turbines and Power,* vol. 124, no. 3, pp. 608-616, 2002.

[150] P. Levine, and L. Angello, "Axial Compressor Performance maintenance," *ASME Turbo Expo,* 2005.

[151] T. W. Song, J. L. Sohn, S. T. Ro, "Predictions of the performance degradation of industrial gas turbines due to compressor fouling," *16th International Symposium on Transport Phenomena,* Prague, 2005.

[152] C. B. M. Homji, "Gas Turbine Axial Compressor Fouling: A Unified Treatment of its Effects, Detection, and Control," *International Journal of Turbo and Jet Engines,* vol. 9, no. 4, pp. 311-334, 1992.

[153] F. Seddigh and H. I. H. Saravanamuttoo, "A Proposed Method for Assessing the Susceptibility of Axial Compressors to Fouling," *ASME Journal of Engineering for Gas Turbines and Power,* vol. 113, no. 4, pp. 595-601, 1991.

[154] K. T. Millsaps, J. Baker, and J. S. Patterson, "Detection and Localization of Fouling in a Gas Turbine Compressors From Aerodynamic Measurements," *ASME Turbo Expo,* Vienna, Austria, 2004.

[155] M. V. Zuniga, "Analysis of Gas Turbine Compressor Fouling and Washing Online," *PhD Thesis,* Cranfield University, 2007.

[156] A. Hamed, W. Tabakoff, D. Singh, "Modelling of Compressor Performance Deterioration Due to Erosion," *International Journal of Rotating Machinery,* vol. 4, no. 4, pp. 243-248, 1998.

[157] A. Hamed, W. Tabakoff, R. Rivir, K. Das, and P. Arora, "Turbine Blade Surface Deterioration by Erosion," *Transactions of the ASME,* vol. 127, no. 3, pp. 445-452, 2005.

[158] Q. Zhou, N. Li, X. Chen, A. Yonezu, T. Xu, S. Hui, and D. Zhang, "Water Drop Erosion on Turbine Blades: Numerical Framework and applications," *Materials Transactions,* vol. 49, no. 7, pp. 1606-1615, 2008.

[159] M. Metwally, W. Tabakoff, A. Hamed, "Blade erosion in automative gas turbine engine," *Journal of Engineering for Gas Turbine and Power,* vol. 117, no. 1, pp. 213-218, 1995.

[160] A. Maus, J. C. Sprott, "Neural network method for determining embedding dimension of a time series," *Communications in Nonlinear Science and Numerical Simulation,* vol. 16, no. 8, pp. 3294-3302, 2011.

[161] F. Shilbayeh and M. Z. Iskandarani, "Effect of Hidden Layer Neurons on the Classification of Optical Character Recognition Typed Arabic Numerals," *Journal of Computer Science,* vol. 4, no. 7, pp. 578-584, 2008.

[162] A. Reynaldi, S. Lukas, H. Margaretha, "Backpropagation and Levenberg-Marquardt Algorithm for Training Finite Element Neural Network," *AMSS 6th European Modelling Symposium,* UKSim, 2012.

[163] J. L. Elman, "Finding Structure in Time," *Cognitive Science,* vol. 14, no. 2, pp. 179-211, 1990.

[164] W. P. J. Visser and M. J. Broomhead, "GSP, a generic object-oriented gas turbine simulation environment," *International Gas Turbine and Aeroengine Congress and Exposition,* 2000.

[165] S. M. Camporeale, B. Fortunato, and M. Mastrovito, "A modular code for real time dynamic simulation of gas turbines in SIMULINK," *Journal of Engineering for Gas Turbines and Power,* vol. 128, no. 3, pp. 506-517, 2006.

[166] GSP Development Team, "GSP 11 User Manual," *National Aerospace Laboratory (NLR),* March 2011.

[167] L. Cai, S. Y. Ma, and Y. L. Zhou, "Prediction of SYM-H index during large storms by NARX neural network from IMF and solar wind data," *Annales Geophysicae,* vol. 28, pp. 381-393, 2010.

[168] K. P. Burnham and D. R. Anderson, "Multimodel Inference Understanding AIC and BIC in Model Selection," *Sociological Methods and Research,* vol. 33, no. 2, pp. 261-304, 2004.

[169] D. Posada and T. R. Buckley, "Model Selection and Model Averaging in Phylogenetics: Advantages of Akaike Information Criterion and Bayesian Approaches Over Likelihood Ratio Tests," *Systematic Biology,* vol. 53, no. 5, pp. 793808, 2004.

[170] H. Acquah, "Comparison of Akaike information criterion (AIC) and Bayesian information criterion (BIC) in selection of an asymmetric price relationship," *Development and Agricultural Economics,* vol. 2, no. 1, pp. 16, 2010.

[171] K. Hirose, S. Kawano, S. Konishi, and M. Ichikawa, "Bayesian Information Criterion and Selection of the Number of Factors in Factor Analysis Models," *Journal of Data Science,* vol. 9, no. 2, pp. 243-259, 2011.

[172] Schwarz and E. Gideon, "Estimating the dimension of a model," *Annals of Statistics,* vol. 6, no. 2, pp. 461464, 1978.

[173] P. Pradeep, "Comparison of variable learning rate and Levenberg-Marquardt back-propagation training algorithms for detecting attacks in Intrusion Detection Systems," *International Journal on*

Computer Science and Engineering, vol. 3, no. 11, pp. 3572-3581, 2011.